

Juhani Kyllönen

MICROSOFT ADO.NET OHJELMISTOTUOTANNOSSA

Opinnäytetyö

Kajaanin ammattikorkeakoulu

Tradenomikoulutus

Syksy 2005



**Kajaanin
ammattikorkeakoulu**

OPINNÄYTETYÖ TIIVISTELMÄ

Ala Luonnontieteiden ala	Koulutusohjelma Tietojenkäsittelyn koulutusohjelma
Tekijä(t) Juhani Kyllönen	
Työn nimi Microsoft ADO.NET ohjelmistokehityksessä	
Vaihtoehtoiset ammattiopinnot Ohjelmointi	Ohjaaja(t) Veli-Pekka Piirainen
Aika Syksy 2005	Sivumäärä 32+15
<p>Tiivistelmä</p> <p>Tämän opinnäytetyön tarkoituksena oli tutustua Microsoft .NET -ohjelmointiympäristöön sisältyvään ADO.NET -tietokantarajapintaan käytännön tasolla. Opinnäytetyön teoriaosuus käsittelee ADO.NET:in perustan muodostavien luokkakirjastojen toimintaa. Tarkastelun kohteena ovat yleisimpien luokkien tärkeimmät ominaisuudet ja metodit. Opinnäytetyön käytännön osassa tavoitteena oli ohjelmoida liiketoimintalogiikka Microsoft SQL -tietokantaa käyttävää ASP.NET -sovellusta varten.</p> <p>ADO.NET on uusin versio ohjelmistotalo Microsoft:n tietokantarajapinnasta. Sen edellisiä versioita ovat olleet mm. ODBC, OLE DB ja ADO. Uusimman version myötä erityyppisten tietokantojen käyttö on helpompaa ja myös tietokantayhteyksien käyttö on tehokkaampaa. Microsoft .NET -ohjelmointiympäristön myötä myös käytettävä ohjelmointikieli on vapaasti valittavissa laajasta valikoimasta.</p> <p>Opinnäytetyön käytännön osuutena toteutettu liiketoimintalogiikka toimii ASP.NET -sovelluksen ja tietokannan välissä tietojen välittäjänä. Se välittää sovelluksesta lähetetyt hakupyynnöt tietokantaan ja palauttaa saadut tulokset takaisin sovellukselle. Erillisen liiketoimintalogiikan päätarkoitus on vähentää tietokantapalvelimeen otettavien yhteyksien määrää. Tämä toteutuu yhteyksien keskittämisen ansiosta, jolloin suoritettu haku voidaan tallentaa välimuistiin ja hakutulosta voidaan käyttää uudelleen. Liiketoimintalogiikka voi myös toimia erillisellä palvelimella ja näin ollen vähentää tietokantapalvelimen sekä ASP.NET -palvelimen kuormitusta.</p> <p>Opinnäytetyön lähteinä on käytetty aihetta käsittelevää kirjallisuutta sekä Internetistä löytyvää materiaalia. Työssä käsiteltyjen luokkakirjastojen ominaisuudet ja metodit selityksineen on myös kerätty Microsoft:n sivustolta yhdeksi kattavaksi kokonaisuudeksi, joka löytyy opinnäytetyön liitteistä.</p>	
Luottamuksellisuus	Julkinen
Hakusanat	.NET, ADO, ADO.NET
Säilytyspaikka	Kajaanin Ammattikorkeakoulun kirjasto

School Business	Degree programme Data Processing
Author(s) Juhani Kyllönen	
Title Microsoft ADO.NET in Software Development	
Alternative professional studies Programming	Instructor(s) Veli-Pekka Piirainen
Date 2005	Total number of pages 32+15
<p>Abstract</p> <p>The purpose of this final year project was to get a practical overview of the ADO.NET -database interface, which is included in the Microsoft .NET programming environment. The theory phase of the final year project considers the functioning of class libraries, which make up the basis of the ADO.NET. The work pays special attention to the most significant properties and methods of the most common class libraries. In the practical phase of the project the purpose was to create a business logic for an ASP.NET application, which connects to Microsoft SQL database.</p> <p>ADO.NET is the latest version of software house Microsoft's database interface. Its previous versions have been, for instance, ODBC, OLE DB and ADO. Along the latest version, the use of different kind of database types is easier. In addition, the use of database connections is more effective. In the Microsoft .NET programming environment also the programming language to be used can be selected freely from a large variety.</p> <p>In the practical phase the created business logic acts as a data mediator between the ASP.NET application and the database. It mediates queries from the application to the database and returns the results to the application. The main purpose of a separate business logic is to decrease the amount of connections to the database. This happens due to the centralisation of the connections, when queries can be cached and results can be used again. The business logic can also run on a standalone server and decrease the loadings of the database server and the ASP.NET server.</p> <p>The sources of this final year project include literature concerning the topic and material found from the Internet. The properties and methods of the handled class libraries have been collected from Microsoft's home page. They can be found in the enclosures of the final year project.</p>	
Confidentiality status	Public
Keywords	.NET, ADO, ADO.NET
Deposited at	Kajaani Polytechnic Library

SISÄLLYS

1	JOHDANTO	1
2	TIETOJEN HALLINTA	2
3	ADO.NET	4
3.1	Managed Provider	4
3.2	Yhteydettömyys	6
3.3	Connection	7
3.4	Command	9
3.5	Transaction	11
3.6	Dataset	12
3.6.1	DataTable	15
3.6.2	DataColumn	17
3.6.3	DataRow	18
3.6.4	DataRelation	19
3.7	DataAdapter	20
3.8	DataReader	21
3.9	SqlParameter	22
3.10	Tietokantayhteyden muodostaminen	23
3.11	Tietojen haku	24
3.12	Tietojen lisäys, päivitys ja poisto	24
3.13	Tietosidonta	25

4	LIKETOIMINTALOGIIKKA ASP.NET –TIETOKANTAOHJELMAAN	27
5	POHDINTA	32

LÄHTEET

LIITTEET

1 JOHDANTO

ADO.NET on Microsoft .NET –ohjelmistoarkkitehtuuriin sisältyvä tietokantarajapinta. Se mahdollistaa edellisiä Microsoft:in julkaisemia tietokantarajapintoja helpommat yhteydet lukuisiin erilaisiin tietokantoihin. Helpompien yhteyksien ohella se myös tehostaa tietokantayhteyksien käyttöä, säästäten tietokantapalvelimien ja verkkoyhteyksien resursseja.

Tämän opinnäytetyön tavoitteena oli perehtyä ADO.NET:iin ja sen tarjoamiin työkaluihin käytännön tasolla. Työn toimeksianto tuli Kajaanin Ammattikorkeakoululta. Työ toteutettiin tutkimalla aiheesta julkaistua kirjallisuutta ja tekemällä tietokantaohjelma Microsoft ASP.NET:iä ja ADO.NET:iä käyttäen. Tietokantaohjelman sovelluskehitysympäristönä käytettiin Microsoft Visual Studio .NET 2003:a ja ohjelmointikielenä Visual Basic .NET:iä. Tietokantapalvelimena käytettiin Microsoft SQL Server 2000:tta ja tietokantana palvelimelta oletuksena löytyvää NorthWind –tietokantaa.

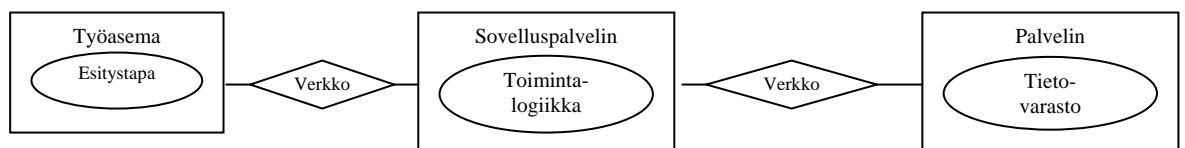
Tietokantaohjelma toteutettiin monitasoarkkitehtuurilla, jossa ohjelman esityskerros ja liiketoimintalogiikka ovat kumpikin erillisten opinnäytetöiden käytännön sovellutukset. Nämä yhdistettynä muodostavat Internet-selaimessa toimivan tietokantaohjelman, jolla voi selata Microsoft SQL –palvelimella olevaa hierarkista tietoa.

2 TIETOJEN HALLINTA

Tietojen hallintajärjestelmä koostuu yksinkertaisimmillaan tietokannasta ja tietokantaa käyttävästä sovelluksesta. Tietokantana voi toimia esimerkiksi Microsoft SQL Server -tietokantapalvelin ja sitä käyttävänä sovelluksena esimerkiksi Internetissä oleva www-sivu. Tällainen kokonaisuus muodostaa mallin, josta käytetään nimitystä asiakas/palvelin- eli kaksitasoarkkitehtuuri. Hieman pidemmälle vietyä tietokannan ja sovelluksen väliin sijoitetaan komponentti, joka välittää tietoja sovelluksen ja tietokannan välillä. Näistä kolmesta sovelluskerroksesta muodostuu malli, josta käytetään nimitystä monitasoarkkitehtuuri.

Monitasoarkkitehtuurilla pyritään vähentämään asiakaskoneessa suoritettavia toimintoja ja tietokantapalvelimeen kohdistuvaa kuormitusta, sijoittamalla sovelluslogiikka omalle palvelimelleen. Asiakaskoneella toimii vain käyttöliittymä, joka kommunikoi sovelluslogiikan kanssa. Jos käyttöliittymään tarvitaan tietoja tietokannasta, sovelluslogiikka muodostaa yhteyden tietokantaan ja suorittaa tarvittavat haut. Myös tietokantaan kohdistuvien yhteyksien määrä vähenee, koska kaikki yhteydet asiakaskoneilta tietokantaan tapahtuvat sovelluslogiikan eli liiketoimintakerroksen kautta. Tällaisesta esityskerroksen, liiketoimintakerroksen ja tietokantakerroksen käsittävästä mallista voidaan käyttää nimitystä kolmitasoarkkitehtuuri (Kuvio 1).

Kuvio 1, kolmitasoarkkitehtuuri



Monitasoarkkitehtuurissa myös esityskerros voidaan sijoittaa erilliselle palvelimelle, jolloin sitä voidaan käyttää asiakaskoneella esimerkiksi Internet-selaimella. Tällaisesta ratkaisusta voidaan käyttää nimitystä nelitasoarkkitehtuuri.

Yksi monitasoarkkitehtuuria hyvin palveleva ohjelmointiympäristö on Microsoft .NET. Se on Microsoftin kehittämä ohjelman ajon aikainen ympäristö, jonka sisällä ohjelma ja sen komponentit suoritetaan. Sen avulla voidaan tehdä esimerkiksi käyttöjärjestelmästä riippumattomia ohjelmia. Ohjelman suorittamiseksi riittää että käyttöjärjestelmään on saatavilla .NET Framework, jonka sisällä ohjelmakoodi suoritetaan.

Yksi Microsoft .NET -ympäristön keskeinen osa on ADO.NET. Se tarjoaa yleisen tietokantaliittymän Microsoft .NET-ympäristössä toimiville ohjelmille ohjelmointikielestä riippumatta. Se on siis rajapinta, jonka kautta voidaan hakea ja ylläpitää dataa erilaisista tietokannoista, kuten Microsoft SQL Server. ADO.NET koostuu tietokantaliittymien hoitamiseen tarkoitetuista luokista, joista käytetään .NET -ympäristössä nimitystä nimiavaruus. (Inkinen 2003, 203-204.)

Microsoft tarjoaa myös helppokäyttöisen työkalun sovelluksien kehittämiseen .NET -ympäristössä, joka on nimeltään Visual Studio .NET. Microsoft Visual Studio .NET on monipuolinen ohjelmointiympäristö, joka sisältää visuaaliset työkalut sovelluskehitykseen. Sen avulla voidaan kehittää ADO.NET:in toimintoja käyttäviä sovelluksia nopeasti ja yksinkertaisesti. Tietokantatoiminnot voidaan toteuttaa sovellukseen käyttämällä ohjelmakoodissa ADO.NET:in luokkia, tai käyttämällä tähän tarkoitukseen tarjolla olevia visuaalisia työkaluja. Työkalujen avulla tietokantojen käsittelyssä tarvittavat toiminnot voidaan lisätä ohjelmaan klikkaamalla kyseisen toiminnon työkalun kuvaketta. Myös toimintojen ominaisuudet voidaan määritellä visuaalisissa asetusikkunoissa.

Microsoft SQL Server on tietokantapalvelin relaatiotietokantojen ylläpitoa varten. Se noudattaa standardia SQL -kyselykieltä, jonka avulla tietokantaa voidaan käyttää asiakasohjelmasta käsin.

3 ADO.NET

ADO.NET on Microsoft .NET Framework -ohjelmointiympäristöön sisältyvä tietokantarajapinta. Se on Microsoftin uusin versio yleiskäyttöisestä tietokantarajapinnasta, jonka edellisiä versioita ovat olleet ODBC, OLE DB ja ADO. Edellisistä versioista poiketen ADO.NET tarjoaa ohjelmoijalle suoran yhteyden lukuisiin eri tietokantatyyppeihin. Tietokantana voi käyttää esimerkiksi Microsoft SQL Server 2000 –tietokantaa, tai mitä tahansa muuta ODBC:tä tai OleDb:tä tukevaa tietokantaa. Tämän ansiosta tietokantaa käyttävissä ohjelmissa ei tarvitse enää käyttää erillisiä tietokanta-ajureita, kuten ODBC-ajureita. Sen sijaan käytettävän tietokannan voi määritellä suoraan ohjelmaan, tai esimerkiksi erilliseen config-tiedostoon. Yksinkertaisimmillaan riittää, että yhteyden asetukset määritellään ohjelmakoodissa ConnectionString –nimiseen Connection –luokan ominaisuuteen. (Martin, Tomson 2002, 259.)

ADO.NET –tietokantarajapinnan keskeisin osa on System.Data –nimiavaruus. Se muodostaa ADO.NET –järjestelmän ytimen. Edellä mainitun lisäksi Microsoft .NET Framework 1.1 :n mukana seuraa myös tietokantakohtaisia nimiavaruuksia, joiden kautta ADO.NET tarjoaa tuen lukuisille eri tietokannoille. (Martin, Tomson 2002, 260.)

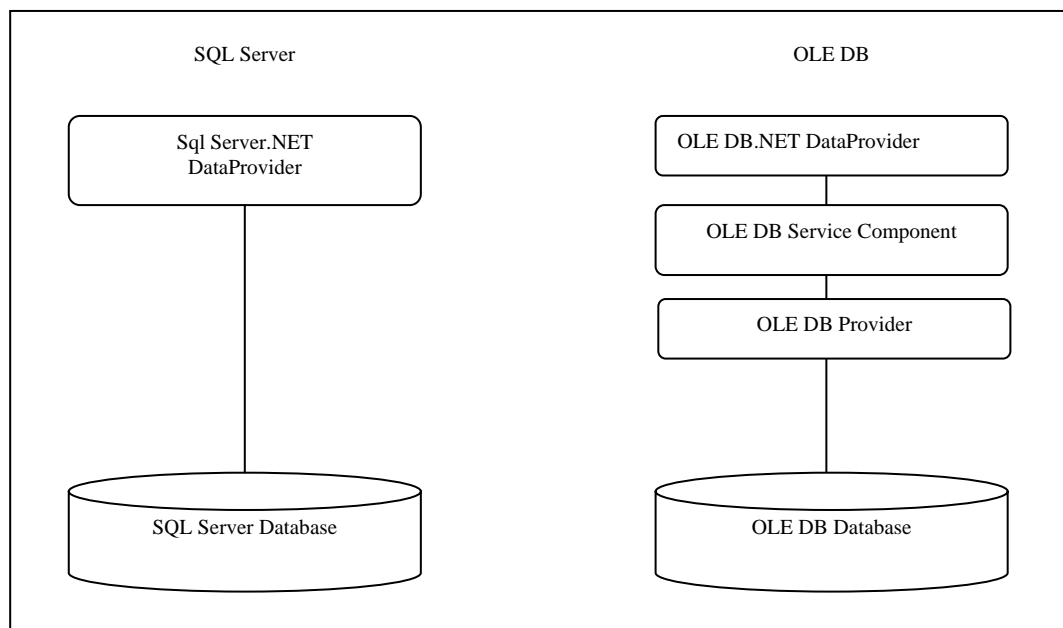
3.1 Managed Provider

Microsoft .NET framework 1.1 sisältää neljä erillistä nimiavaruutta, joiden kautta voidaan käyttää eri tyyppisiä tietokantoja. Jokainen nimiavaruus kapseloi sisäänsä tietojen välittäjä-objektin, josta käytetään nimitystä Data Provider. Microsoft .NET Framework 1.1 sisältää omat Data Provider:it Microsoft SQL Server tietokannoille versiosta 7.0 lähtien, OleDb:tä tukeville tietokannoille, ODBC:tä tukeville tietokannoille sekä Oracle -tietokannoille. (MSDN 2005 a.)

Microsoft .NET Framework 1.1 :n tietokantakohtaiset nimiavaruudet ovat:

- System.Data.SqlClient
- System.Data.OleDb
- System.Data.Odbc
- System.Data.OracleClient.

System.Data.SqlClient kapseloi Data Provider:in Microsoftin omaa SQL Server –tietokantaa varten. Se on optimoitu käytettäväksi pelkästään MS SQL Server –yhteyksiin ja tarjoaa sen ansiosta yli kaksi kertaa nopeamman yhteyden kuin OleDb:n kautta (Inkinen 2003, 204). SqlConnection –Data Provider on yhteen sopiva Microsoft SQL Server –tietokantoihin versiosta 7.0 lähtien. Suoritusnopeutta on kasvatettu karsimalla Data Providerin välikerroksia. Kuvio 2 havainnollistaa SqlConnection- ja OleDb –providereiden eroja.



Kuvio 2, SqlConnection ja OleDb -luokkien ero (Inkinen 2003, 205)

System.Data.OleDb kapseloi Data Provider:in OleDb:tä tukevien tietokantojen käyttöä varten. Näitä ovat esimerkiksi Oracle, Access-tietokanta, Excel-tilukko, tekstitiedosto ja Microsoft SQL Server 6.0. OleDb:tä käytettäessä täytyy määrittellä myös käytettävän tietokannan tyyppi. Tämä määrittellään lisäämällä Connection -objektin ConnectionString -ominaisuuteen Provider -ominaisuus ja antamalla sille tietokantaa vastaava arvo.

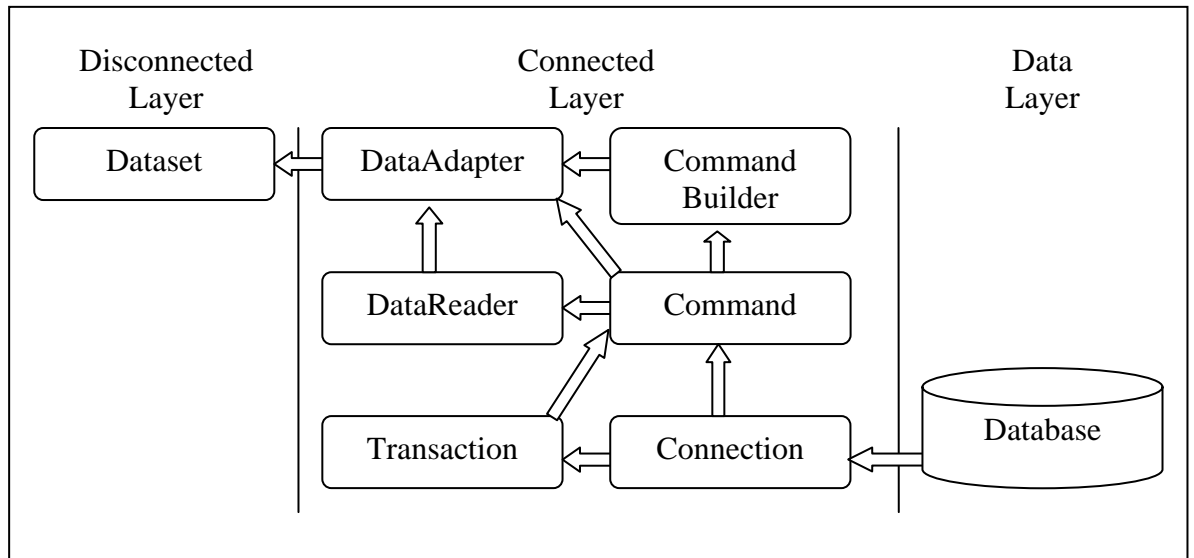
System.Data.Odbc kapseloi Data Provider:in ODBC –yhteen sopivia tietokantoja varten. Myös sitä käytettäessä täytyy Connection –objektin ConnectionString –ominaisuuteen lisätä Provider –ominaisuus ja sen tietokantakohtainen arvo.

System.Data.OracleClient kapseloi sisäänsä Data Provider:in Oracle –tietokantoja varten. Se on yhteensopiva Oracle –asiakasohjelmien kanssa versiosta 8.1.7 lähtien.

Yllä mainittujen lisäksi ADO.NET:iin on saatavilla myös lukuisia muiden ohjelmistovalmistajien tuottamia Data Provider:eita. Joitakin tietokantoja varten on esimerkiksi kehitelty optimoituja Data Provider:eita, joiden avulla tietokantayhteyksistä saadaan tehokkaampia kuin Microsoft:in tarjoamilla yleisillä Provider:eilla. Data Provider:eita on kehitelty myös sellaisille tietokannoille, joille ei ole tukea Microsoft .NET Framework 1.1:ssä. (SQLSummit.com 2005.)

3.2 Yhteydettömyys

Yksi ADO.NET:in keskeisistä ominaisuuksista on yhteydettömyys. Lähemmin tarkasteltuna yhteydettömyys koskee DataSet –objektia, jolla ei ole suoraa yhteyttä varsinaisiin tietokantoihin. Tiedot varsinaisesta tietokannasta voidaan hakea DataSet –objektiin yhteydellisen DataAdapter –objektin kautta. Haetut tiedot tallentuvat asiakaskoneen muistissa olevaan DataSet –objektiin ja niitä voidaan muokata yhteydettömästi. DataSet –objektin muokatut tiedot voidaan vastaavasti syöttää tietokantaan DataAdapter –objektin kautta. Kuvio 3 havainnollistaa ADO.NET:in rakennetta ja yhteydettömyyttä.



Kuvio 3, ADO.NET ja yhteydettömyys (Elca 2005)

3.3 Connection

Connection on ADO.NET:in perusluokka, jonka avulla rakennetaan yhteys haluttuun tietokantaan. Luokasta on omat versionsa jokaiselle Data Provider:ille, esimerkiksi MS SQL Server –yhteyksiä varten oleva luokka on nimeltään SqlConnection. OleDb –yhteyksiä varten oleva luokka taas on yleiskäyttöinen OleDbConnection -luokka. (Martin, Tomson 2002, 264.)

Connection -objektin tärkein ominaisuus on ConnectionString, johon asetetaan tietokantayhteyteen tarvittavat parametrit. Yksinkertaisimmillaan tarvittavat parametrit tietokantayhteyden käyttämiseen ovat tietokantapalvelimen osoite, tietokannan nimi, käyttäjätunnus ja salasana. Näiden lisäksi voidaan asettaa erilaisia käyttäjän tunnistukseen ja tietoturvaan liittyviä parametrejä. Taulukossa 1 on lueteltu SqlConnection –luokan ominaisuudet ja niiden selitykset.

Taulukko 1, SqlConnection-luokan ominaisuudet (MSDN 2005 b.)

ConnectionString	Asettaa/palauttaa Microsoft SQL Server –tietokantayhteyden määrittelevän merkkijonon
ConnectionTimeout	Palauttaa yhteydenmuodostusyrityksen odotusajan, jonka jälkeen yritys lopetetaan ja generoidaan virhe
Database	Palauttaa nykyisen tietokannan nimen
DataSource	Palauttaa yhteyteen käytettävän SQL Serverin instanssin nimen
PacketSize	Palauttaa SQL Serverin instanssin kanssa kommunikointiin käytettävien tietoliikennepakettien koon tavuissa
ServerVersion	Palauttaa yhdistettynä olevan SQL Serverin instanssin version kertovan merkkijonon
State	Palauttaa nykyisen tietokantayhteyden tilan
WorkstationId	Palauttaa tietokanta-asiakkaan identifioivan merkkijonon

Connection –objekti sisältää muutamia metodeja, joista tärkeimmät ovat Connection.Open ja Connection.Close. Connection.Open –metodi avaa yhteyden tietokantaan. Kun yhteyttä ei enää tarvita, se suljetaan Connection.Close –metodilla. SqlConnection –luokan metodit ja niiden selitykset on lueteltu taulukossa 2.

Taulukko 2, SqlConnection-luokan metodit (MSDN 2005 b.)

BeginTransaction	Aloittaa tietokantatapahtuman
ChangeDatabase	Vaihtaa avoimen SQL-yhteyden käyttämän tietokannan
Close	Sulkee tietokantayhteyden
CreateCommand	Luo ja palauttaa SqlConnection-objektiin yhdistetyn SqlCommand-objektin
Open	Avaa ConnectionString-ominaisuudessa määritellyn tietokantayhteyden

SqlConnection –luokka sisältää lisäksi kaksi julkista tapahtumaa, jotka ovat InfoMessage ja StateChange. Ensin mainittu tapahtuu, kun SQL -palvelin palauttaa varoituksen tai informatiivisen viestin. Jälkimmäinen tapahtuu, kun yhteyden tila muuttuu. (MSDN 2005 b.)

EsimerkkejäConnectionString –merkkijonosta:

1) MS SQL Server:

```
"Data Source=Aron1;Initial Catalog=pubs;User Id=sa;Password=;"
(ConnectionStrings.com)
```

2) Access:

```
"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=path\db.mdb;User Id=admin;Password=;"
(ConnectionStrings.com)
```

3) Oracle:

```
"Provider=MSDAORA;Data Source=MyOracleDB;User Id=User;Password=;"
(ConnectionStrings.com)
```

3.4 Command

Command on ADO.NET:in objekti, jota käytetään kyselyjen välittämiseen tietokantaan. Objektista on Data Provider –kohtaiset versiot, joista MS SQL Server –tietokantaa varten oleva on nimeltään SqlCommand. OleDb –yhteyksiä varten tarjolla oleva taas käyttää nimeä OleDbCommand. (Martin, Tomson 2002, 266.)

SqlCommand -luokan ja OleDbCommand -luokan tärkeimmät ominaisuudet ovat CommandText ja Connection. CommandText -ominaisuuteen määritellään kyselyn hakulauseke. Connection -ominaisuuteen taas määritellään käytettävä tietokantayhteys. (Martin, Tomson 2002, 266.)

Muita kummallakin luokalla olevia tärkeämpiä ominaisuuksia ovat CommandType, Parameters ja Transaction. CommandType määrittelee CommandText-ominaisuuden tyyppin, eli mitä ominaisuus pitää sisällään. Vaihtoehtoina ovat talletetun proseduurin nimi,

kokonaisen palautettavan taulun nimi tai suoritettava SQL -lause. Parameters-ominaisuus määrittelee parametrikokoelman, joka sisältää muuttujia sisältävän SQL -lauseen muuttujaparametrit. Transaction-ominaisuus määrittelee/palauttaa tietokantatapahtuman, jonka avulla voidaan varmistaa komentojen täydellinen suorittaminen. SqlCommand -luokan julkiset ominaisuudet ja niiden selitykset on lueteltu taulukossa 3. (Martin, Tomson 2002, 266.)

Taulukko 3, SqlCommand-luokan julkiset ominaisuudet (MSDN 2005 c.)

CommandText	Asettaa/palauttaa suoritettavan SQL-hakulauseen tai tietolähteesä suoritettavan talletetun proseduurin
CommandTimeout	Asettaa/palauttaa odotusajan, jonka jälkeen yritys suorittaa komento lopetetaan ja generoidaan virhe
CommandType	Asettaa/palauttaa arvon, joka määrää kuinka CommandText-ominaisuus tulkitaan
Connection	Asettaa/palauttaa kyseisessä SqlCommand-instanssissa käytettävän SqlConnection-objektin
Parameters	Palauttaa SqlParameterCollection-objektin
Transaction	Asettaa/palauttaa SqlTransaction-objektin, jonka sisällä komento suoritetaan
UpdatedRowSource	Asettaa/palauttaa kuinka komennon tuloksia sovelletaan DataRow-objektiin DbDataAdapter-objektin Update-metodia käytettäessä

SqlCommand- ja OleDbcommand -luokkien tärkeimmät suoritukseen liittyvät metodit ovat Cancel, ExecuteNonQuery ja ExecuteReader. Cancel -metodilla voidaan keskeyttää komennon suoritus. ExecuteNonQuery suorittaa komennon, joka ei palauta tietokannasta mitään. Tällainen voi olla esimerkiksi tietojen lisäys tietokantaan. ExecuteReader suorittaa hakukomennon lähettämällä kyselyn Connection -objektille ja muodostaa DataReader-objektin, jonka kautta tulokset palautuvat. SqlCommand -luokan julkiset metodit ja niiden selitykset on lueteltu taulukossa 4. (Martin, Tomson 2002, 266.)

Taulukko 4, SqlCommand-luokan julkiset metodit (MSDN 2005 c.)

Cancel	Yrittää keskeyttää SQL-käskyn suorittamisen
CreateParameter	Luo uuden instanssin SqlParameter –objektista
ExecuteNonQuery	Suorittaa SQL-käskyn ja palauttaa siihen kohdistuneiden rivien määrän
ExecuteReader	Lähetää CommandText-ominaisuuden Connection-objektille ja muodostaa SqlDataReader-objektin
ExecuteScalar	Suorittaa kyselyn ja palauttaa tulosten ensimmäisen rivin ensimmäisen sarakkeen.
ExecuteXmlReader	Lähetää CommandText-ominaisuuden Connection-objektille ja muodostaa XmlReader-objektin
Prepare	Muodostaa valmistellun käskyn SQL Serverin instanssilla
ResetCommandTimeout	Palauttaa CommandTimeout-ominaisuuden oletusarvoonsa

3.5 Transaction

Transaction on System.Data.SqlClient –nimiavaruuteen sisältyvä objekti, jonka avulla voidaan varmistaa useiden peräkkäisten tietokantakomentojen toteutuminen. Tästä on hyötyä esimerkiksi tilanteessa, jossa suoritetaan useita tietokannan päivityskomentoja silmukan sisällä. Komennot suoritetaan Transaction –objektin Commit –metodin avulla ja tapahtuman keskeytyessä tehdyt muutokset voidaan perua RollBack –metodin avulla. Näin voidaan varmistaa, että tietokannan päivitys ei jää virheen sattuessa osittaiseksi. (Standardio.org.)

SqlTransaction-luokka sisältää julkiset ominaisuudet Connection ja IsolationLevel. Connection –ominaisuus palauttaa transaktioon liitetyn SqlConnection –objektin ja IsolationLevel määrittelee transaktion eristämistason. SqlTransaction –luokan julkiset metodit ja niiden selitykset on lueteltu taulukossa 5. (MSDN 2005 d.)

Taulukko 5, SqlTransaction –luokan julkiset metodit (MSDN 2005 d.)

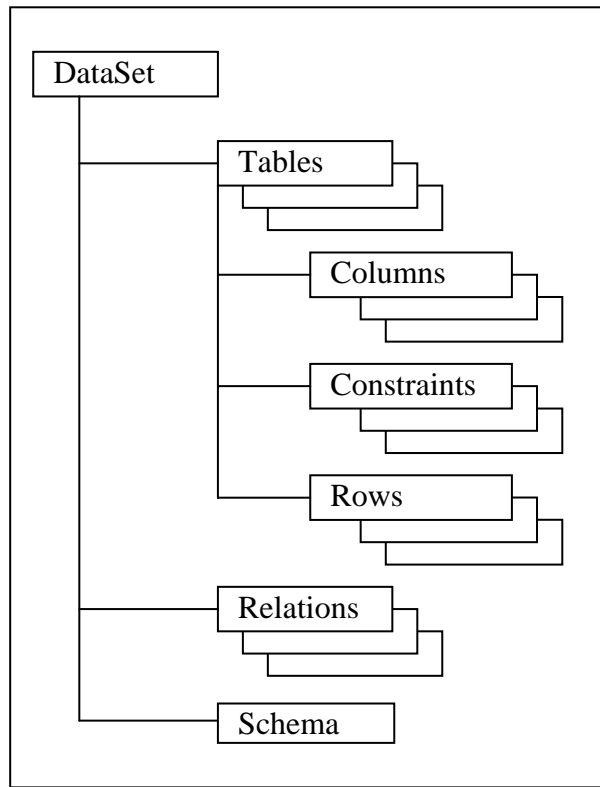
Commit	Aloittaa transaktion suorittamisen
Dispose	Vapauttaa SqlTransaction –objektin käyttämät resurssit
RollBack	Peruuttaa transaktion suoritusta edeltäneeseen tilaan
Save	Luo tallennuspisteen transaktiossa, jota voidaan käyttää transaktion osittaiseen peruuttamiseen

3.6 Dataset

Dataset on System.Data –nimiavaruuteen sisältyvä tärkein tietojenkäsittelyobjekti. Se on yleiskäyttöinen tietokannan rakennetta vastaava, yhteydetön tietojen säilytysobjekti. Siihen voidaan tallettaa tietoa tietokannan tauluja vastaavassa muodossa, mukaan lukien taulujen väliset viiteavainyhteydet ja kenttien sisältöä rajoittavat ehdot. Yhteydettömyyden ansiosta Datasetin sisältämien taulujen muuttaminen ei päivity suoraan tietolähteen tauluihin, mutta muutokset voidaan haluttaessa päivittää tietolähteeseen DataAdapter -objektin avulla. (Inkinen 2003, 209.)

Dataset koostuu DataTable -objekteista, jotka vastaavat rakenteeltaan tietokannan tauluja. DataTable taas koostuu DataRow- ja DataColumn -objekteista, jotka vastaavat relaatiotaulun riviä ja saraketta. DataRow- ja DataColumn -objektien avulla Dataset -objektiin voidaan tallettaa tietoja ja muokata niitä, aivan kuten relaatiotietokannassa. Tiedot tallentuvat tietokoneen muistiin XML-muodossa. Taulujen viiteavainyhteyksien kuvaamiseen Dataset-objektissa käytetään DataRelations -objektia, johon voidaan tallettaa taulujen väliset yhteydet. Kuvio 4 kuvaa DataSet -objektin rakennetta. (Martin, Tomson 2002, 298.)

Kuviossa alimpana osana näkyvä Schema, eli suomeksi skeema, on tekniikka rakenteen kuvaukseen. Se kuvaa DataSet –objektin rakenteen XML –muodossa.



kuvio 4, Dataset-objektin rakenne (Elca 2005.)

DataSet-luokan tärkeimmät julkiset ominaisuudet ovat DataSetName, Relations ja Tables. DataSetName –ominaisuus asettaa tai palauttaa DataSet –objektin nimen. Relations –ominaisuus palauttaa kokoelman yhteyksiä, jotka linkittävät tauluja ja mahdollistavat navigoinnin isätauluista lapsitauluihin. Tables –ominaisuus taas palauttaa DataSet -objektin sisältämät taulut kokoelmana. Eräitä Muita Dataset-luokan julkisia ominaisuuksia ja niiden selitykset on lueteltu taulukossa 6. Täydellinen lista ominaisuuksista löytyy opinnäytetyön liitteistä. (MSDN 2005 e.)

Taulukko 6, Dataset-luokan julkisia ominaisuuksia (MSDN 2005 e.)

CaseSensitive	Asettaa/palauttaa, erotteleeko merkkijonovertailu isot ja pienet kirjaimet toisistaan DataTable-objektin sisällä
DefaultViewManager	Palauttaa Dataset-objektin tiedot määrättyinä näkymänä sallien suodatuksen, etsinnän ja selaamisen
EnforceConstraints	Asettaa/palauttaa noudatetaanko viite-eheyssääntöjä tietoja päivittäessä
ExtendedProperties	Palauttaa Dataset-objektiin liitetyt räätälöidyt tiedot
HasErrors	Palauttaa Dataset-objektin DataTable-objekteissa olevien virheid ^{en} määrän

Dataset-luokan tärkeimpiä julkisia metodeja ovat AcceptChanges, RejectChanges ja Clear –metodit. AcceptChanges –metodi hyväksyy kaikki Dataset –objektiin sen latauksen tai edellisen AcceptChanges –metodin kutsun jälkeen tehdyt muutokset. RejectChanges vastaa-
vasti peruuttaa kaikki Dataset –objektiin sen muodostamisen tai Dataset.AcceptChanges –
metodin kutsun jälkeen tehdyt muutokset. Clear –metodi tyhjentää Dataset –objektin pois-
tamalla kaikki rivit kaikista sen sisältämistä tauluista. Muita Dataset –luokan julkisia meto-
deja ja niiden selitykset on lueteltu taulukossa 7. Täydellinen taulukko metodeista löytyy
opinnäytetyön liitteistä. (MSDN 2005 e.)

Taulukko 7, Dataset-luokan julkisia metodeja (MSDN 2005 e.)

Clone	Kopioi Dataset-objektin rakenteen, mukaan lukien DataTable- objektien skeemat, yhteydet ja rajoitteet. Ei kopioi dataa.
GetXml	Palauttaa Dataset-objektiin talletetun datan XML –muodossa
GetXmlSchema	Palauttaa XSD-skeeman Dataset-objektiin talletetun datan XML –muodolle
HasChanges	Palauttaa tiedon, onko Dataset-objektiin tehty muutoksia, mu- kaan lukien rivien lisäys, poisto tai muuttaminen
ReadXml	Lukee XML-skeeman ja tiedot Dataset-objektiin
Reset	Asettaa Dataset-objektin alkuperäiseen tilaansa.
WriteXml	Kirjoittaa tiedot ja tarvittaessa myös skeeman Dataset-objektista

DataSet-luokka sisältää yhden julkisen tapahtuman, joka on nimeltään MergeFailed. Se tapahtuu, kun kohteena ja lähteenä olevissa DataRow -objekteissa on sama pääavaimen arvo ja EnforceConstraints -ominaisuuden arvo on tosi. (MSDN 2005 e.)

3.6.1 DataTable

DataTable -objekti on tiedon tallennusobjekti, joka vastaa rakenteeltaan relaatiotietokannan taulua. Se sisältää kokoelmat DataColumnCollection ja DataRowCollection. Näistä ensiksi mainittu pitää sisällään taulun sarakkeet eli kentät ja jälkimmäinen vastaavasti rivit eli tietueet. (Martin, Tomson 2002, 301.)

DataTable -objektin rakenne määräytyy DataColumnCollection -objektista, joka muodostuu DataColumn -objekteista. DataColumn -objektit sisältävät taulun kentät ja niiden tietotyypit. Varsinainen tieto puolestaan sisältyy DataRowCollection -objektiin, joka muodostuu DataRow -objekteista. DataRow -objekti vastaa relaatiotietokannan tietuetta, eli siihen talletetaan kutakin taulun kenttää vastaavat arvot. DataRowCollection -objekti säilyttää tiedoista myös niiden alkuperäisiä arvoja, joiden avulla tietoihin tehdyt muutokset voidaan perua. (Martin, Tomson 2002, 301.)

DataTable -luokka sisältää 17 julkista ominaisuutta, joista olennaisimmat selityksineen on lueteltu taulukossa 8. Täydellinen taulukko luokan julkisista ominaisuuksista ja niiden selityksistä löytyy opinnäytetyön liitteistä.

Taulukko 8, DataTable-luokan julkisia ominaisuuksia (MSDN 2005 f.)

ChildRelations	Palauttaa DataTable-objektin lapsitaulu-yhteydet kokoelmana
Columns	Palauttaa DataTable-objektiin kuuluvat sarakkeet kokoelmana
Constraints	Palauttaa DataTable-objektin ylläpitämät rajoitukset kokoelmana
Dataset	Palauttaa Dataset-objektin, johon DataTable-objekti kuuluu
ParentRelations	Palauttaa DataTable-objektin isätaulu-yhteydet kokoelmana
PrimaryKey	Asettaa/palauttaa taulun pääavainkenttinä olevat sarakkeet
Rows	Palauttaa tauluun kuuluvat rivit kokoelmana
TableName	Asettaa/palauttaa DataTable-objektin nimen

DataTable-luokka sisältää 18 julkista metodia, joista osa selityksineen on lueteltu taulukossa 9. Täydellinen taulukko luokan metodeista ja niiden selityksistä löytyy opinnäytetyön liitteistä. DataTable-luokka sisältää myös kuusi julkista tapahtumaa, jotka on lueteltu selityksineen taulukossa 10.

Taulukko 9, DataTable-luokan julkisia metodeja (MSDN 2005 f.)

AcceptChanges	Hyväksyy kaikki tauluun edellisen AcceptChanges-metodin kutsun jälkeen tehdyt muutokset
Clear	Tyhjentää DataTable-objektista kaikki tiedot
Copy	Kopioi DataTable-objektin rakenteen ja tiedot
GetChanges	Palauttaa kopion DataTable-objektista, sisältäen latauksen tai AcceptChanges-metodin kutsun jälkeen tehdyt muutokset
GetErrors	Palauttaa taulukon virheitä sisältävistä DataRow-objekteista
NewRow	Luo uuden DataRow-objektin, jonka rakenne on sama kuin taulun
RejectChanges	Peruuttaa kaikki tauluun muodostamisen tai AcceptChanges-metodin kutsun jälkeen tehdyt muutokset
Reset	Palauttaa DataTable-objektin alkuperäiseen tilaansa

Taulukko 10, DataTable-luokan julkiset tapahtumat (MSDN 2005 f.)

ColumnChanged	Tapahtuu, kun DataRow-objektin määrätyn DataColumn-objektin arvo on muuttunut
ColumnChanging	Tapahtuu, kun DataRow-objektin määrätyn DataColumn-objektin arvo on muuttumassa
RowChanged	Tapahtuu, kun DataRow-objekti on muutettu onnistuneesti
RowChanging	Tapahtuu, kun DataRow-objekti on muuttumassa
RowDeleted	Tapahtuu, kun taulun rivi on poistettu
RowDeleting	Tapahtuu, kun taulun riviä ollaan poistamassa

3.6.2 DataColumn

DataColumn on objekti, joka määrittelee DataTable -objektin yhden sarakkeen ja sen tietotyytin. DataTable -objektin rakenne eli skeema koostuu yhdestä tai useammasta siihen lisätyistä DataColumn -objektista. (Martin, Tomson 2002, 305.)

DataColumn -luokka sisältää 18 julkista ominaisuutta, joihin kuuluvat esimerkiksi DataType, Table ja Unique -ominaisuudet. DataType -ominaisuus asettaa tai palauttaa sarakkeen tietotyytin ja Table -ominaisuus palauttaa DataTable -objektin, johon sarake kuuluu. Unique -ominaisuus asettaa tai palauttaa arvon, joka määrää pitääkö sarakkeen rivien arvojen olla yksilöllisiä. Taulukossa 11 on listattu luokan muita tärkeimpiä ominaisuuksia selityksineen. Täydellinen taulukko luokan ominaisuuksista selityksineen löytyy opinnäytetyön liitteistä. (MSDN 2005 g.)

Taulukko 11, DataColumn –luokan julkisia ominaisuuksia (MSDN 2005 g.)

AllowDBNull	Asettaa/palauttaa arvon, joka määrää voiko sarake sisältä null – arvoja
AutoIncrement	Asettaa/palauttaa arvon, joka määrää lisääkö uuden rivin sarakkeeseen automaattisesti arvo
Caption	Asettaa/palauttaa sarakkeen otsikon
ColumnName	Asettaa/palauttaa sarakkeen nimen
DefaultValue	Asettaa/palauttaa sarakkeen oletusarvon
Namespace	Asettaa/palauttaa DataColumn -objektin nimiavaruuden
ReadOnly	Asettaa/palauttaa arvon, joka määrää voiko sarakkeen arvoja muuttaa tauluun lisäämisen jälkeen

DataColumn –luokalla on lisäksi julkinen ToString –metodi, joka palauttaa sarakkeen Expression –ominaisuuden, jos se on määritelty (MSDN 2005 g.).

3.6.3 DataRow

DataRow on objekti, joka sisältää DataTable -objektin yhdelle riville tallennetut tiedot. DataRow toimii yhdessä DataColumn -objektin kanssa, joka määrittelee DataRow -objektin sarakkeiden tietotyypit. DataRow -objekti sisältää viisi rivin käsittelemiseen liittyvää julkista ominaisuutta, jotka on lueteltu selityksineen taulukossa 12. (Martin, Tomson 2002, 306.)

Taulukko 12, DataRow –luokan julkiset ominaisuudet (MSDN 2005 h.)

HasErrors	Palauttaa arvon, joka kertoo onko rivillä virheitä
ItemArray	Asettaa/palauttaa rivin kaikki arvot merkkijonon kautta
RowError	Asettaa/palauttaa riville räätälöidyn virheilmoituksen
RowState	Palauttaa rivin nykyisen tilan
Table	Palauttaa DataTable-objektin, jonka skeemaa rivi noudattaa

DataRow –objektilla on 16 julkista metodia, joista tärkeimpiä selityksineen on lueteltu taulukossa 13. Täydellinen lista metodeista löytyy opinnäytetyön liitteistä.

Taulukko 13, DataRow –luokan julkisia metodeja (MSDN 2005 h.)

AcceptChanges	Hyväksyy kaikki riviin edellisen AcceptChanges –metodin kutsun jälkeen tehdyt muutokset
BeginEdit	Aloittaa DataRow –objektin muokkauksen
CancelEdit	Keskeyttää nykyisen rivin muokkauksen
Delete	Poistaa rivin
EndEdit	Lopettaa rivin muokkauksen
GetChildRows	Palauttaa rivin lapsirivit
GetParentRow	Palauttaa rivin isärivin
RejectChanges	Peruuttaa kaikki riviin edellisen AcceptChanges –metodin kutsun jälkeen tehdyt muutokset

3.6.4 DataRelation

DataRelation -objektin avulla voidaan liittää kaksi samaan DataSet -objektiin sisältyvää DataTable -objektia toisiinsa. Liitos on samankaltainen kuin Microsoft SQL relaatiotietokannassa, eli yhteys asetetaan DataTable -objektin DataColumn -objektista toisen DataTable -objektin samansisältöiseen DataColumn -objektiin. (Martin, Tomson 2002, 308.)

DataRelation -objekti huolehtii myös viite-eheyksien säilymisestä. Se peruuttaa tapahtuman ja aiheuttaa poikkeustilanteen, jos riville lisätään tietoa sopimattomassa muodossa. (Martin, Tomson 2002, 308.)

Taulukko 14, DataRelation -luokan julkiset ominaisuudet (MSDN 2005 i.)

ChildColumns	Palauttaa yhteyden lapsisarakkeet
ChildTable	Palauttaa yhteyden lapsitaulun
DataSet	Palauttaa DataSet -objektin, johon DataRelation -objekti kuuluu
ParentColumns	Palauttaa DataRelation -objektin isäsarakeet
ParentTable	Palauttaa DataRelation -objektin isätaulun
RelationName	Asettaa/palauttaa nimen, jolla DataRelation -objekti voidaan löytää DataRelationCollection -kokoelmasta

DataRelation -luokalla on julkinen ToString -metodi, joka palauttaa RelationName -ominaisuuden, jos se on määritelty. Lisäksi luokalla on suojattu CheckStateForProperty -metodi, joka varmistaa että yhteysobjekti on kelvollinen. (MSDN 2005 i.)

3.7 DataAdapter

DataAdapter on ADO.NET:in yhteydellinen objekti, jonka kautta yhteydettömät DataSet- ja DataTable -objektit saavat yhteyden tietokantaan. Taulukossa 15 on lueteltu DataAdapter -objektin julkiset ominaisuudet. Näiden lisäksi tietokantakohtaisilla DataAdapter -objekteilla on omia ominaisuuksia. Esimerkiksi SqlDataAdapter -objekti sisältää neljä omaa ominaisuutta, jotka ovat DeleteCommand, InsertCommand, SelectCommand ja UpdateCommand (MSDN 2005 n.). DataAdapter –objektin julkiset metodit selityksineen on lueteltu taulukossa 16.

Taulukko 15, DataAdapter-luokan julkiset ominaisuudet (MSDN 2005 j.)

AcceptChangesDuringFill	Asettaa/palauttaa arvon, joka määrää kutsutaanko AcceptChanges –metodia kun rivi lisätään DataTable -objektiin
ContinueUpdateOnError	Asettaa/palauttaa arvon, joka määrää generoidaanko poikkeus jos rivin päivityksessä tapahtuu virhe
MissingMappingAction	Määrittelee suoritettavan toimenpiteen, jos syötteenä tulevalle datalle ei ole sopivaa taulua tai saraketta
MissingSchemaAction	Määrittelee suoritettavan toimenpiteen, jos DataSet -objektin skeema ei sovi syötteenä tulevalle datalle
TableMappings	Palauttaa kokoelman, joka huolehtii lähdetaulun ja DataTable –objektin välisestä kartoituksesta

Taulukko 16, DataAdapter-luokan julkiset metodit (MSDN 2005 j.)

Fill	Lisää tai päivittää DataSet-objektin rivit vastaamaan tietolähteen rivejä ja luo DataTable-objektin nimeltä ”Table”
FillSchema	Lisää määrittäytyn DataSet-objektiin ”Table” –nimisen DataTable-objektin ja muotoilee skeeman vastaamaan tietolähdettä
GetFillParameters	Palauttaa käyttäjän asettamat parametrit SQL Select –lausetta varten
Update	Kutsuu asianomaisia INSERT, UPDATE tai DELETE –lauseita jokaiselle lisätylle, muutetulle tai poistetulle riville määrättyssä DataSet -objektissa

3.8 DataReader

DataReader-objekti tarjoaa DataAdapter -objektia nopeamman tavan hakea tietokannasta tietoja. Se on tarkoitettu pelkästään tietojen hakuun, eikä sen kautta voi päivittää tietoja tietokantaan. DataReader:in kautta tapahtuva tietokannan tietueiden luku tapahtuu aina alusta loppuun päin. Se sopii erityisen hyvin tilanteisiin, joissa tietokantaan kohdistetaan useita peräkkäisiä pyyntöjä. Tällaisia voivat olla esimerkiksi dynaamiset www-sivut, joilla näytettävä tieto haetaan tietokannasta hakuehtojen mukaan. DataReader -objektilla on kuusi julkista ominaisuutta, jotka palauttavat tietoa objektin toiminnasta ja sisällöstä. Ne on lueteltu selityksineen taulukossa 18. (Martin, Tomson 2002, 268.)

Taulukko 18, DataReader –luokan julkiset ominaisuudet (MSDN 2005 k.)

Depth	Palauttaa nykyisen rivin sisäkkäisyyden syvyyden kertovan arvon
FieldCount	Palauttaa sarakkeiden määrän nykyisessä rivissä
HasRows	Palauttaa arvon, joka kertoo SqlDataReader-objektissa olevien rivien määrän
IsClosed	Palauttaa arvon, joka kertoo onko tiedon lukija suljettu
Item	Palauttaa sarakkeen arvon sen natiivissa muodossa
RecordsAffected	Palauttaa SQL-käskyn suorituksen yhteydessä muutettujen, syötettyjen tai poistettujen rivien määrän

DataReader -objekti sisältää 40 julkista metodia, joista suurin osa liittyy objektin palauttamien tietojen tietotyyppeihin. Taulukossa 19 on lueteltu objektin tärkeimmät metodit, sekä muutama tietotyyppeihin liittyvä metodi. Täydellinen lista metodeista löytyy opinnäytetyön liitteistä. (MSDN 2005 k.)

Taulukko 19, SqlDataReader –luokan julkisia metodeja (MSDN 2005 k.)

Close	Sulkee SqlDataReader-objektin
GetBoolean	Palauttaa määrätyn sarakkeen arvon boolean-muodossa
GetDateTime	Palauttaa määrätyn sarakkeen arvon DateTime-objektina
GetInt32	Palauttaa määrätyn sarakkeen arvon 32 -bittisenä kokonaisluku- na
GetName	Palauttaa määrätyn sarakkeen nimen
GetValues	Palauttaa nykyisen rivin arvolliset sarakkeet kokoelmana
NextResult	Ohjaa readerin seuraavaan tulokseen, kun luetaan tuloksia erä- ajona suoritettavista SQL-käskyistä
Read	Ohjaa SqlDataReader-objektin seuraavaan tietueeseen

3.9 SqlParameter

SqlParameter -objektin avulla muuttujia sisältävien SQL -lauseiden muuttujat voidaan sijoittaa SqlCommand -objektin Parameters -kokoelmaan. Tämä helpottaa muuttujien käyttämistä monimutkaisissa SQL -lauseissa ja parantaa myös suoritusnopeutta. (Martin, Tomson 2002, 284.)

Suoritusnopeuden kasvuun vaikuttaa parametroidin tietotyyppimäärittäminen. Toisin kuin tavallisessa SQL -lauseessa, parametrisoidussa SQL -lauseessa tietotyyppitieto käännetään suoraan ASP.NET -koodiksi. Tämän ansiosta SQL -palvelimen ei tarvitse tehdä tietotyyppitystä ajon aikana ja suoritusnopeus paranee. Muita parametrisoinnin tarjoamia etuja ovat tietojen helpompi tarkistus, koodin parempi luettavuus sekä olioperusteisuuden ansiosta parempi joustavuus. (Martin, Tomson 2002, 284.)

Taulukko 20, SqlParameter-luokan julkisia ominaisuuksia (MSDN 2005 l.)

IsNullable	Asettaa/palauttaa arvon, joka määrää hyväksyykö parametri nolla- arvoja
ParameterName	Asettaa/palauttaa SqlParameter -objektin nimen
Precision	Asettaa/palauttaa desimaalien maksimimäärän Value - ominaisuuden esittämisessä
Scale	Asettaa/palauttaa desimaalien määrän, johon Value -ominaisuus pyöristetään
Size	Asettaa/palauttaa kentässä olevan tiedon maksimimäärän tavuina
Value	Asettaa/palauttaa parametrin arvon

SqlParameter-luokka sisältää myös julkisen ToString -metodin, joka palauttaa ParameterName -ominaisuuden merkkijonona.

3.10 Tietokantayhteyden muodostaminen

Tietokantayhteys ADO.NET:in avulla voidaan muodostaa joko määrittelemällä se ohjelmakoodissa tai käyttämällä Microsoft Visual Studio yhteystyökalua. Näistä jälkimmäinen tapa on opastavan käyttöliittymänsä ansiosta yksinkertaisempi.

Muodostettaessa yhteyttä ohjelmakoodissa, luodaan uusi ilmentymä käytettävän Data Provider:in Connection-objektista. Jos yhteys halutaan muodostaa Microsoft SQL Server -tyyppiseen tietokantaan, objektina on järkevintä käyttää SqlConnection -objektia sen optimointien ansiosta. Muissa tapauksissa yhteys voidaan muodostaa esimerkiksi OleDbConnection tai OracleConnection -objektin kautta. Seuraavaksi objektista luodulle ilmentymälle annetaan tarvittavat yhteysparametrit sijoittamalla ne objektin ConnectionString-ominaisuuteen ja yhteys on valmis käytettäväksi.

Käytettäessä Microsoft Visual Studiosta löytyvää työkalua, valitaan ensin työkalulaatikon data -välilehdeltä käytettävään tietokantaan sopiva työkalu ja vedetään se lomakkeelle. Klikkaamalla lomakkeella näkyvän työkalun Data/ConnectionString -ominaisuuden tekstikenttää ja siihen ilmestyvää nuolta, päästään muodostamaan uusi tietokantayhteys. Ruudulle avautuu lomake, jonka avulla valitaan tietokantapalvelin sekä käytettävä tietokanta ja syötetään tarvittavat käyttäjätunnukset. Lomakkeelta löytyy myös käytettävissä olevat OLE DB -palvelut, toiminto yhteyden testaamiseen, ohjeen avaava painike sekä muutamia yhteyteen vaikuttavia lisäasetuksia.

Tietokantayhteyden avaaminen koodissa tapahtuu Connection -luokan Open -metodilla. Tietokantayhteyttä käytettäessä tulee huolehtia, ettei yhteyttä pidetä turhaan auki. Jokainen avoin yhteys kuormittaa SQL -palvelinta, jonka vuoksi yhteyksien määrä on järkevintä pitää mahdollisimman pienenä. Kun yhteyden kautta suoritettavat SQL -lauseet on suoritettu, se suljetaan Connection -luokan Close -metodilla.

3.11 Tietojen haku

Tietojen haku tietokannasta tapahtuu Command -objektiin sijoitettavien hakulausekkeiden avulla. Tietokantahaku valmistellaan luomalla objektista uusi ilmentymä ja määrittelemällä siihen hakulauseke sekä käytettävä tietokantayhteys. Mikäli tietokannasta haetaan yhtä tiettyä hakuehdot täyttävää tulosta, Command -objekti suoritetaan sen ExecuteScalar -metodilla. Jos taas halutaan kaikki ehdot täyttävät tulokset, suorittamiseen käytetään ExecuteReader -metodia. (Martin, Tomson 2002, 267.)

Hakulausekkeena tietojen haussa käytetään haun kohteena olevan tietokannan tukemaa hakusyntaksia. Esimerkiksi Microsoft SQL Server-, Oracle- ja Microsoft Access -tietokannoista tietoa haettaessa käytetään SQL-lauseita. SQL on standardien mukainen kyselykieli, joka perusmuodossaan toimii yleisimmissä tietokannoissa. Jotkut tietokantavalmistajat ovat tehneet tietokantoihinsa kuitenkin myös ominaisuuksia, joita käytetään standardista poikkeavilla SQL -lauseilla. Tällaisia ominaisuuksia sisältyy muun muassa Oracle -tietokantaan. Perustoiminnot kuitenkin toimivat normaaleilla SQL -lauseilla. Hakulause voidaan sijoittaa etukäteen erilliseen string-tyyppiseen muuttujaan, tai kirjoittaa suoraan Command-objektiin. Myös käytettävä tietokantayhteys on mahdollista määritellä suoraan Command-objektiin, mutta selkeyden ja uudelleen käytettävyyden vuoksi sekä hakulause että tietokantayhteys on parempi määritellä omiin string-tyyppisiin muuttujiinsa.

Jos tietokantahakuja tehdään esimerkiksi www-sivulle syötettävien hakusanojen perusteella, SQL -lauseisiin sijoitettavat muuttujat voidaan sijoittaa SqlCommand -objektin Parameters -kokoelmaan. Se selkeyttää hakulauseiden rakennetta ja tehostaa myös hakujen suoritusta parametroidun tietotyyppimäärityksen ansiosta.

3.12 Tietojen lisäys, päivitys ja poisto

Tietojen lisäyksiä, muokkauksia ja poistoja voidaan suorittaa yksinkertaisimmillaan tietojen haun tapaan sijoittamalla SQL -lause ja käytettävä tietokantayhteys Command -objektiin.

Useita muuttujia sisältävien lauseiden yhteydessä voidaan myös käyttää Command -objektin Parameters -kokoelmaa. Command -objekti suoritetaan sen ExecuteNonQuery -metodilla, koska suoritetaan SQL -lause joka ei palauta tuloksia.

Suoritettaessa kerralla useita päivityskomentoja, päivitysten täydellinen toteutuminen voidaan varmistaa transaktiolla. Sen avulla voidaan perua tehdyt muutokset, jos päivitystapah-tuma keskeytyy virheeseen.

Tietojen lisäys, päivitys ja poisto voidaan toteuttaa myös DataAdapter -objektin Update -metodilla. Sitä voidaan käyttää tilanteessa, jossa tiedot DataSet -objektiin on haettu tietokannasta. Kun DataSet:in sisältämiä tietoja on muokattu, muutokset voidaan välittää DataAdapter -objektin kautta lähdetietokantaan. DataAdapter analysoi DataSet:iin tehdyt muutokset ja suorittaa muuttuneiden tietojen osalta tarvittavat INSERT, UPDATE ja DELETE -komennot tietokantaan. (MSDN 2005 m.)

3.13 Tietosidonta

Microsoft .NET -ympäristö sisältää myös komponentteja, jotka voivat käyttää suoraan tietokantayhteyttä sisältönsä muodostamiseen. Tällaisia ovat muun muassa ASP.NET -sivuilla käytettäviksi tarkoitettut DataGrid ja DataList web-palvelinkontrollit. Niiden avulla tietoa voidaan tulostaa www-sivulle helposti taulukkomuodossa. Yksi tällaisten komponenttien yhteinen piirre on DataSource -ominaisuus, johon määritellään tietolähde komponentin täyttöä varten. Tietolähteenä voidaan käyttää esimerkiksi ExecuteReader -metodilla suoritettua SqlCommand -objektia. Tämä nopeuttaa tietojen hakua ja esittämistä, koska tietoja ei tarvitse tallettaa DataTable -objektiin. Varsinainen tietolähteen liittäminen komponenttiin suoritetaan komponentin DataBind -metodilla. Tässäkin tapauksessa tulee kuitenkin huolehtia erikseen tietokantayhteyden avaamisesta ja sulkemisesta. Yhteys avataan ennen SqlCommand -objektin suorittamista ja suljetaan DataBind -metodin kutsumisen jälkeen. (Martin, Tomson 2002, 277-279; Inkinen 2003, 220)

DataGrid web-palvelinkontrolli tarjoaa myös helpon tavan tietojen muokkaukseen. Se sisältää muokkaustilan, jossa taulukon tietoja voidaan muokata rivi kerrallaan. Päivitetyt tiedot voidaan myös välittää edelleen tietokantaan DataAdapter -objektin kautta. Tässä tapauksessa tietokannan tiedot täytyy kuitenkin hakea DataGrid -kontrollille DataTable -objektin kautta. DataGrid -kontrollille tehdyt muutokset päivitetään ensin DataTable -objektiin, josta DataAdapter -objekti voi välittää muutokset edelleen tietokantaan DataTable -objektin säilyttämien alkuperäisten tietojen perusteella.

4 LIIKETOIMINTALOGIIKKA ASP.NET -TIETOKANTAOHJELMAAN

Opinnäytetyön käytännön osuutena toteutettiin ADO.NET:iä käyttäen liiketoimintakomponentti web-selaimessa toimivaan ASP.NET -tietokantasovellukseen. Sovelluskehittimenä käytettiin Microsoft Visual Studio .NET 2003:a ja sovelluskehityskielenä Visual Basic .NET:iä. Kajaanin ammattikorkeakoulun opiskelija Timo Nevala teki ASP.NET -tietokantasovelluksen osana omaa opinnäytetyötään. Liiketoimintakomponentti toimii sovelluksen ja Microsoft SQL Serverin välissä tietojenvälittäjänä. Se toimittaa sovelluksesta lähetetyt kyselyt SQL Serverille ja palautetut tiedot edelleen sovellukselle.

Liiketoimintakomponentin suunnittelu aloitettiin tutustumalla monitasohierarkiaan käytännön tasolla. Tutustumisen kohteena oli monitasoarkkitehtuuria noudattava .NET -ympäristössä toimiva tietokantaohjelma, joka on toteutettu käyttäen C# -ohjelmointikieltä.

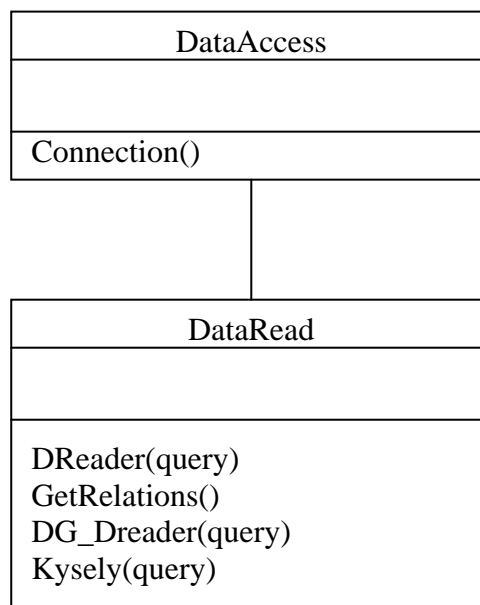
Työssä käyttämäni tietokoneen käyttöjärjestelmänä toimi Windows 2000. Käytännön työn ensimmäinen vaihe oli asentaa tietokoneelle työssä tarvittavat ohjelmat ja palvelimet. Ne olivat MS Visual Studio .NET 2003, MS SQL Server 2000 -tietokantapalvelin ja MS IIS www-palvelin. Myöhemmin sovelluksen testaamista varten täytyi vielä asentaa sovelluksen ASP.NET -sivulla käytettävä MS TreeView -komponentti.

Ohjelmien asennuksessa ilmeni muutamia ongelmia, jotka liittyivät lähinnä .NET Framework:in määrittelyyn IIS:ssä ja SQL Server 2000:n käyttäjien tunnistukseen. .NET Framework täytyi rekisteröidä käytettäväksi IIS:ssä suorittamalla komentokehoteessa aspnet_regiis.exe /i -komento. Kyseinen tiedosto löytyy Windows -hakemiston alla olevan Microsoft.NET\Framework\v1.1.4322 -polun päästä.

MS SQL Server 2000 asetti asennuksen yhteydessä autentikoinnin vain Windows-käyttäjätunnuksille, jonka vuoksi palvelimen käyttö itsemääritellyillä käyttäjätunnuksilla ei onnistunut. Ongelman ratkaisu oli asettaa palvelimen tietoturva-asetuksista autentikointimuodoksi ”sql server and windows”.

Myös TreeView -kontrollin käytössä ilmeni ajoittain ongelmia. Se sisältää www-selaimessa näytettäviä kuvakkeita ja niiden toimintaan saaminen oli välillä hankalaa. Kuvakkeet sijaitsivat IIS -palvelimen Inetpub\wwwroot -hakemistossa ja hakemiston polku täytyy määritellä tarkasti TreeView -kontrollin ominaisuuksiin.

Tietokantaohjelman nimeksi tuli SQLServerBrowser, jonka vuoksi looginen valinta liiketoimintakomponentin nimeksi oli SQLServerBrowser.BusinessLogic. Liiketoimintalogiikka koostuu kahdesta luokasta, jotka ovat DataAccess.vb ja DataRead.vb. Liiketoimintalogiikan luokkarakenne on kuvattu kuviossa 5. Liiketoimintakomponentti saadaan käyttöön ASP.NET -sovelluksessa lisäämällä se sovelluksen referensseihin. Tämän jälkeen komponentin DataRead -luokasta luodaan ilmentymä, jonka kautta luokan funktioita voidaan kutsua. DataAccess -luokka sisältää pelkästään Connection() -nimisen julkisen jaetun funktion, johon on määritelty käytettävä tietokantayhteys. DataRead -luokan funktiot käyttävät sitä tietokantayhteyksien muodostamiseen.



Kuvio 5, Liiketoimintalogiikan luokkakaavio

DataRead -luokka sisältää kaikki tietojen välittämisessä ASP.NET -sivun ja tietokannan välillä tarvittavat funktiot. Nämä ovat nimeltään Dreader, GetRelations, DG_Dreader ja Kysely.

Dreader on funktio, joka palauttaa ASP.NET -sovelluksen TreeView -kontrollissa tarvittavat tietokannan taulujen nimet. Se saa lähtötietoina string -tyyppisen muuttujan ja suorittaa sen perusteella tietokantahaun käyttäen SqlDataReader -objektia. Tämän jälkeen funktio muodostaa ilmentymän ArrayList -objektista ja täyttää sen SqlDataReader -objektin ensimmäisen sarakkeen tiedoilla. Tämä sarake sisältää tietokannan taulujen nimet. Lopuksi ArrayList palautetaan ASP.NET -sovellukselle.

GetRelations on funktio, joka palauttaa tietokannan taulujen väliset relaatiot ASP.NET -sovellukselle DataTable -objektina. Se suorittaa SQL -lauseen, joka palauttaa relaatiot SqlDataReader -objektin kautta. Tämän jälkeen relaatiot sijoitetaan DataTable -objektiin, joka palautetaan ASP.NET -sovellukselle.

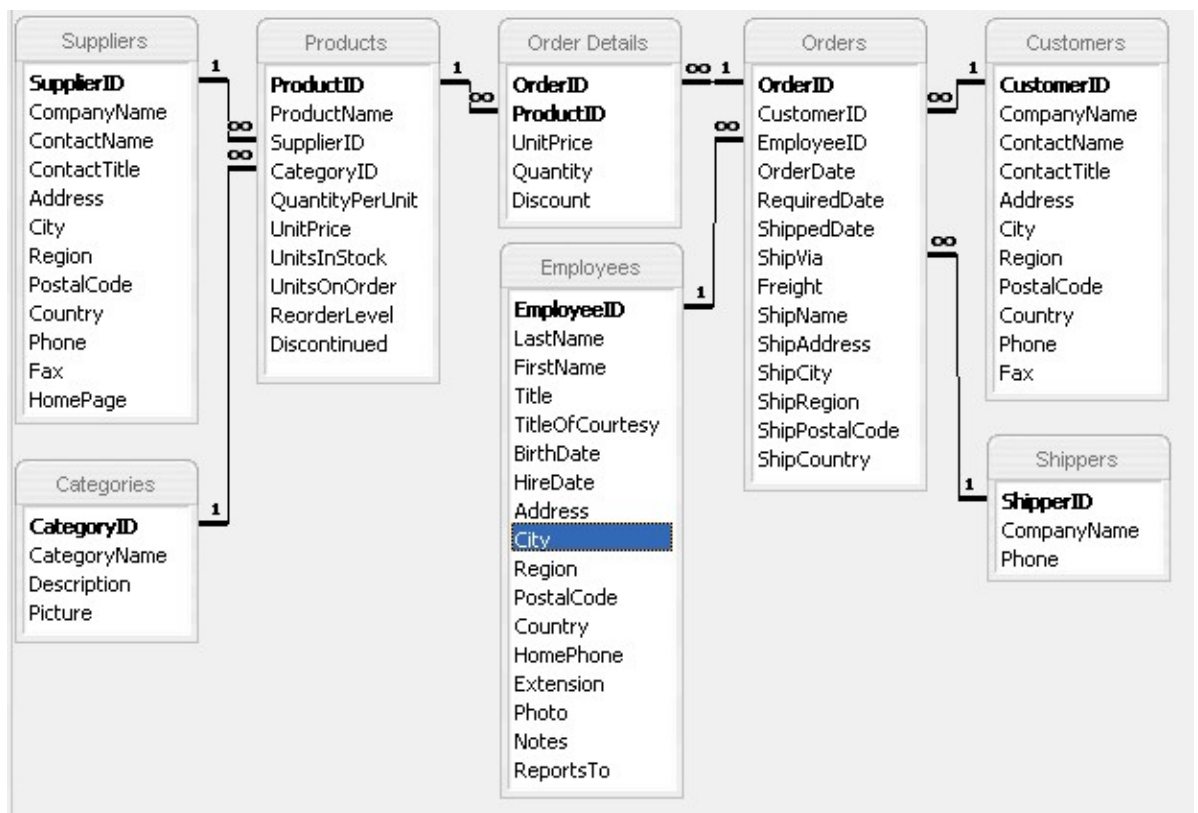
DG_Dreader on funktio, joka hakee ASP.NET -sivun DataGrid -kontrollilla näytettävät tiedot. Se saa lähtötietoina string -tyyppisen muuttujan, joka sisältää DataGrid -kontrollilla näytettävän taulun nimen. Funktio suorittaa taulun nimen perusteella tietokantahaun, joka hakee kyseisen taulun kaikki tietueet. Funktio käyttää hakemiseen DataReader -objektia ja palauttaa DataReader -objektin täytettynä ASP.NET -sovellukselle.

Kysely on funktio, joka saa lähtötietona string -tyyppisen muuttujan. Muuttuja sisältää SQL -lauseen, jonka funktio suorittaa tietokantaan. Funktio muodostaa uuden ilmentymän DataSet -objektista ja täyttää sen lähtötietona saamansa string -tyyppisen muuttujan perusteella. Lopuksi se palauttaa täytetyn DataSet -objektin ASP.NET -sovellukselle. ASP.NET -sovellus käyttää tätä funktiota muun muassa taulujen nimien hakemisessa TreeView -kontrolliin sekä DataGrid -kontrollin otsikon nimen hakemisessa.

Käytännön työ aloitettiin suunnittelemalla Timo Nevalan opinnäytetyönä toteutettavan ASP.NET -sivun ulkoasu ja kartoittamalla sen tarvitsemat liiketoimintakomponentit. Varsinaisen liiketoimintalogiikan koodaamista aloitettaessa tehtiin pieni WinForm -sovellus

logiikan testaamista varten. Työn edetessä testisovellukseksi jouduttiin kuitenkin vaihtamaan Nevalan tekemä ASP.NET -sovellus, koska WinForm -sovelluksessa käytettävien kontrollien ominaisuudet ja metodit poikkeavat jonkin verran WebForm -sovelluksessa käytettävistä.

ASP.NET -sovelluksen toiminnallinen tavoite oli pystyä selaamaan Microsoft SQL -palvelimella olevaa hierarkista tietoa web -selaimella. Lähtökohtana oli tehdä tietokanta-riippumaton selain, eli taulujen nimiä tai tietueiden kenttien järjestystä ei koodattaisi itse ohjelmaan. Testitietokantana käytettiin Microsoft SQL Server 2000 -tietokantapalvelimelta oletuksena löytyvää NorthWind -mallitietokantaa, joka vastaa lähinnä normaalin päivittäistavarakaupan tietokantaa. Tietokanta sisältää muun muassa taulut Categories, Employees, Order Details, Orders ja Products. NorthWind -tietokannan rakenne on kuvattu kuviossa 6. Sovellusta testattiin onnistuneesti myös Microsoft SQL Server 2000 -tietokantapalvelimelta oletuksena löytyvällä Pubs -tietokannalla.



Kuvio 6, NorthWind -tietokannan rakenne

ASP.NET -sovellus näyttää tietokannassa olevat taulut selaimen vasemmassa laidassa puunäkymänä. Taulun nimen vasemmalla puolella olevaa plus -merkkiä painamalla puu laajenee taulun kohdalta ja taulun nimen alle ilmestyy kyseisen taulun lapsitaulut. Samalla selaimen oikeaan laitaan tulee näkyviin taulun tietueet taulukkomuodossa. Lapsitaulua painamalla taulukon sisällöksi vaihtuu lapsitaulun tietueet. Puu myös jälleen laajenee lapsitaulun plus -merkkiä painamalla, jos lapsitaululla on omia lapsitauluja. Kuviossa 7 näkyy ruutu-kaappaus asp-sivusta ja puun toiminnasta.

Orders

		OrderID	CustomerID	EmployeeID	OrderDate
Edit	Delete	10248	VINET	5	4.7.1996 04:00
Update	Delete	10249	TOMSP	6	5.7.1996 00:00
Cancel					
Edit	Delete	10250	HANAR	4	8.7.1996 04:00
Edit	Delete	10251	VICTE	3	8.7.1996 04:00
Edit	Delete	10252	SUPRD	4	9.7.1996 04:00
Edit	Delete	10253	HANAR	3	10.7.1996 04:00
Edit	Delete	10254	CHOPS	5	11.7.1996 04:00
Edit	Delete	10255	RICSU	9	12.7.1996 04:00
Edit	Delete	10256	WELLI	3	15.7.1996 04:00
Edit	Delete	10257	HILAA	4	16.7.1996 04:00
Edit	Delete	10258	ERNSH	1	17.7.1996 04:00
Edit	Delete	10259	CENTC	4	18.7.1996 04:00
Edit	Delete	10260	OTTIK	4	19.7.1996 04:00
Edit	Delete	10261	QUEDE	4	19.7.1996 04:00
Edit	Delete	10262	RAITC	8	22.7.1996 04:00
Edit	Delete	10263	ERNSH	9	23.7.1996 04:00
Edit	Delete	10264	FOLKO	6	24.7.1996 04:00
Edit	Delete	10265	BLONP	2	25.7.1996 04:00
Edit	Delete	10266	WARTH	3	26.7.1996 04:00
Edit	Delete	10267	FRANK	4	29.7.1996 04:00
Edit	Delete	10268	GROSR	8	30.7.1996 04:00

Orders [Edit] [Delete]

Kuvio 7, kuva käyttöliittymästä

5 POHDINTA

Microsoft .NET on vielä suhteellisen tuore ohjelmointiympäristö ja se tulee varmasti yleistyään lähivuosien aikana. Aiheeseen keskittyvä koulutuskin on ollut ainakin Kajaanin ammattikorkeakoulussa vielä melko vähäistä ja kurssitarjonta tulee toivottavasti kasvamaan tulevaisuudessa.

Opinnäytetyön tekeminen oli haasteellinen ja erittäin opettavainen prosessi. Aiheesta ei juurikaan ollut pohjatietoja, vaan Microsoft ADO.NET täytyi opiskella aihetta käsitteleviä julkaisuja lukemalla ja käytännön kokeiluilla. Asiaa käsittelevä kirjallisuus oli myöskin vielä melko vähäistä ja Internetissä aihetta käsittelevistä keskustelupalstoista oli paljon hyötyä. Microsoft .NET -ohjelmoinnista oli jo ennestään kolmen opintoviikon kurssi suoritettuna, mutta se keskittyi Microsoft .NET:in ja C# -ohjelmointikielen perusteisiin. Opinnäytetyön käytännön osa taas toteutettiin käyttäen Microsoft Visual Studio .NET -ohjelmointikieltä. Myös IIS -palvelun käyttöönotto, web-sivujen toteutus Microsoft ASP.NET:llä ja monitasoarkkitehtuuri tulivat käytännön osuutta tehtäessä tutuiksi.

Opinnäytetyön tekeminen syvensi aiempia opintojani Microsoft.NET –arkkitehtuurista ja loi hyvän pohjan osallistua tulevaisuudessa .NET –ohjelmistoprojekteihin. Aihetta käsittelevien aineistojen tutkiminen herätti kiinnostusta myös muiden Microsoft .NET –arkkitehtuurin tarjoamien työkalujen, sekä tulevan Microsoft .NET Framework 2.0:n mukana tulevien uudistuksien itsenäiseen opiskeluun.

LÄHTEET

- Inkinen Ville, 2003. ASP.NET. Jyväskylä: Docendo Finland Oy.
- Martin J., Tomson B., 2002. ASP.NET Trainer Kit. Helsinki: Edita Prima Oy.
- Elca 2005 <http://www.elca.ch/resources/ADO.NET.pdf> (luettu 10.3.2005).
- MSDN 2005 a.
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpguide/html/cpconusingadonetproviderstoaccessdata.asp> (luettu 12.1.2005)
- MSDN 2005 b.
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfsystemdatasqlclientsqlconnectionmemberstopic.asp> (luettu 12.1.2005)
- MSDN 2005 c.
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfsystemdatasqlclientsqlcommandmemberstopic.asp> (luettu 20.1.2005)
- MSDN 2005 d.
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfsystemdatasqlclientsqltransactionmemberstopic.asp> (luettu 26.1.2005)
- MSDN 2005 e.
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfsystemdatadatadatasetmemberstopic.asp> (luettu 6.2.2005)
- MSDN 2005 f.
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfsystemdatadatatablememberstopic.asp> (luettu 20.2.2005)
- MSDN 2005 g.
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfsystemdatadatadacolumnmemberstopic.asp> (luettu 20.2.2005)
- MSDN 2005 h.
<http://msdn.microsoft.com/library/en-us/cpref/html/frlrfsystemdatadatatarowmemberstopic.asp> (luettu 5.3.2005)
- MSDN 2005 i.
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfsystemdatadatarelationmemberstopic.asp> (luettu 5.3.2005)

MSDN 2005 j.

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfssystemdatacomondataadaptermemberstopic.asp>
(luettu 27.3.2005)

MSDN 2005 k.

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfssystemdatasqlclientsqldataareadermemberstopic.asp>
(luettu 27.3.2005)

MSDN 2005 l.

<http://msdn.microsoft.com/library/en-us/cpref/html/frlrfssystemdatasqlclientsqlparametermemberstopic.asp>
(luettu 4.4.2005)

MSDN 2005 m.

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpguide/html/cpconupdatingdatabewithdataadapterdataset.asp>
(luettu 12.4.2005)

MSDN 2005 n.

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfssystemdatasqlclientsqldataadaptermemberstopic.asp>
(luettu 22.4.2005)

ConnectionStrings.com 2005 <http://www.connectionstrings.com> (luettu 12.1.2005)

Standardio.org 2005 <http://www.standardio.org/article.aspx?id=168> (Luettu 14.1.2005)

SQLSummit.com 2005 <http://www.sqlsummit.com/DataProv.htm> (luettu 23.4.2005)

LIITTEET

- Liite 1: SqlConnection -luokan julkiset ominaisuudet, metodit ja tapahtumat
- Liite 2: SqlCommand -luokan julkiset ominaisuudet ja metodit
- Liite 3: SqlTransaction -luokan julkiset ominaisuudet ja metodit sekä Dataset -luokan julkiset ominaisuudet
- Liite 4: Dataset -luokan julkiset metodit ja tapahtumat
- Liite 5: DataTable -luokan julkiset ominaisuudet
- Liite 6: DataTable -luokan julkiset metodit
- Liite 7: DataTable -luokan julkiset tapahtumat
- Liite 8: DataColumn -luokan julkiset ominaisuudet
- Liite 9: DataRow -luokan julkiset ominaisuudet ja metodit
- Liite 10: DataRelation -luokan julkiset ominaisuudet
- Liite 11: DataAdapter -luokan julkiset ominaisuudet ja metodit
- Liite 12: DataReader -luokan julkiset ominaisuudet ja metodit
- Liite 13: SqlParameter -luokan julkiset ominaisuudet

Taulukko 1, SqlConnection -luokan julkiset ominaisuudet (MSDN 2005 b.)

ConnectionString	Asettaa/palauttaa Microsoft SQL Server –tietokantayhteyden määrittävän merkkijonon
ConnectionTimeout	Palauttaa yhteydenmuodostusyrityksen odotusajan, jonka jälkeen yritys lopetetaan ja generoidaan virhe
Database	Palauttaa nykyisen tietokannan nimen
DataSource	Palauttaa yhteyteen käytettävän SQL Serverin instanssin nimen
PacketSize	Palauttaa SQL Serverin instanssin kanssa kommunikointiin käytettävien tietoliikennepakettien koon tavuissa
ServerVersion	Palauttaa yhdistettynä olevan SQL Serverin instanssin version kertovan merkkijonon
State	Palauttaa nykyisen tietokantayhteyden tilan
WorkstationId	Palauttaa tietokanta-asiakkaan identifioivan merkkijonon

Taulukko 2, SqlConnection -luokan julkiset metodit (MSDN 2005 b.)

BeginTransaction	Aloittaa tietokantatapahtuman
ChangeDatabase	Vaihtaa avoimen SQL-yhteyden käyttämän tietokannan
Close	Sulkee tietokantayhteyden
CreateCommand	Luo ja palauttaa SqlConnection-objektiin yhdistetyn SqlCommand-objektin
Open	Avaa ConnectionString-ominaisuudessa määritellyn tietokantayhteyden

Taulukko 3, SqlConnection -luokan julkiset tapahtumat (MSDN 2005 b.)

InfoMessage	Tapahtuu, kun SQL palvelin palauttaa varoituksen tai informatiivisen viestin
StateChange	Tapahtuu, kun yhteyden tila muuttuu

Taulukko 4, SqlCommand -luokan julkiset ominaisuudet (MSDN 2005 c.)

CommandText	Asettaa/palauttaa suoritettavan SQL-hakulauseen tai tietolähteessä suoritettavan talletetun proseduurin
CommandTimeout	Asettaa/palauttaa odotusajan, jonka jälkeen yritys suorittaa komento lopetetaan ja generoidaan virhe
CommandType	Asettaa/palauttaa arvon, joka määrää kuinka CommandText-ominaisuus tulkitaan
Connection	Asettaa/palauttaa kyseisessä SqlCommand-instanssissa käytettävän SqlConnection-objektin
Parameters	Palauttaa SqlParameterCollection-objektin
Transaction	Asettaa/palauttaa SqlTransaction-objektin, jonka sisällä komento suoritetaan
UpdatedRowSource	Asettaa/palauttaa kuinka komennon tuloksia sovelletaan DataRow-objektiin DbDataAdapter-objektin Update-metodia käytettäessä

Taulukko 5, SqlCommand -luokan julkiset metodit (MSDN 2005 c.)

Cancel	Yrittää keskeyttää SQL-käskyn suorittamisen
CreateParameter	Luo uuden instanssin SqlParameter -objektista
ExecuteNonQuery	Suorittaa SQL-käskyn ja palauttaa siihen kohdistuneiden rivien määrän
ExecuteReader	Lähetää CommandText-ominaisuuden Connection-objektille ja muodostaa SqlDataReader-objektin
ExecuteScalar	Suorittaa kyselyn ja palauttaa tulosten ensimmäisen rivin ensimmäisen sarakkeen.
ExecuteXmlReader	Lähetää CommandText-ominaisuuden Connection-objektille ja muodostaa XmlReader-objektin
Prepare	Muodostaa valmistellun käskyn SQL Serverin instanssilla
ResetCommandTimeout	Palauttaa CommandTimeout-ominaisuuden oletusarvoonsa

Taulukko 6, SqlConnection -luokan julkiset ominaisuudet (MSDN 2005 d.)

Connection	Palauttaa transaktioon liitetyn SqlConnection -objektin
IsolationLevel	Määrittelee transaktion eristämistason

Taulukko 7, SqlConnection -luokan julkiset metodit (MSDN 2005 d.)

Commit	Aloittaa transaktion suorittamisen
Dispose	Vapauttaa SqlConnection -objektin käyttämät resurssit
RollBack	Peruuttaa transaktion suoritusta edeltäneeseen tilaan
Save	Luo tallennuspisteen transaktiossa, jota voidaan käyttää transaktion osittaiseen peruuttamiseen

Taulukko 8, DataSet -luokan julkiset ominaisuudet (MSDN 2005 e.)

CaseSensitive	Asettaa/palauttaa, erotteleeko merkkijonovertailu isot ja pienet kirjaimet toisistaan DataTable-objektin sisällä
DataSetName	Asettaa/palauttaa DataSet-objektin nimen
DefaultViewManager	Palauttaa DataSet-objektin tiedot määrättyinä näkymänä sallien suodatuksen, etsinnän ja selaamisen
EnforceConstraints	Asettaa/palauttaa noudatetaanko viite-eheyssääntöjä tietoja päivitettäessä
ExtendedProperties	Palauttaa DataSet-objektiin liitetyt räätälöidyt tiedot
HasErrors	Palauttaa DataSet-objektin DataTable-objekteissa olevien virheiden määrän
Locale	Asettaa/palauttaa merkkijonojen vertailussa taulun sisällä vaikuttavan lokaalitiedon
Namespace	Asettaa/palauttaa DataSet-objektin nimiavaruuden
Prefix	Asettaa/palauttaa DataSet-objektin nimiavaruuden peitenimenä käytettävän XML:n etuliitteen
Relations	Palauttaa kokoelman yhteyksiä, jotka linkittävät tauluja ja mahdollistavat navigoinnin isätauluista lapsitauluihin
Tables	Palauttaa DataSet-objektin sisältämät taulut kokoelmana

Taulukko 9, Dataset -luokan julkiset metodit (MSDN 2005 e.)

AcceptChanges	Hyväksyy kaikki Dataset-objektiin latauksen tai edellisen hyväksymisen jälkeen tehdyt muutokset
Clear	Tyhjentää Dataset-objektin poistamalla kaikki rivit kaikista tauluista
Clone	Kopioi Dataset-objektin rakenteen, mukaan lukien DataTable-objektien skeemat, yhteydet ja rajoitteet. Ei kopioi dataa.
GetXml	Palauttaa Dataset-objektiin talletetun datan XML –muodossa
GetXmlSchema	Palauttaa XSD-skeeman Dataset-objektiin talletetun datan XML –muodolle
HasChanges	Palauttaa tiedon, onko Dataset-objektiin tehty muutoksia, mukaan lukien rivien lisäys, poisto tai muuttaminen
InferXmlSchema	Hakee XML-skeeman Dataset-objektiin
ReadXml	Lukee XML-skeeman ja tiedot Dataset-objektiin
ReadXmlSchema	Lukee XML-skeeman Dataset-objektiin
RejectChanges	Peruuttaa kaikki Dataset-objektiin muodostamisen tai DataSet.AcceptChanges-metodin kutsun jälkeen tehdyt muutokset
Reset	Asettaa Dataset-objektin alkuperäiseen tilaansa. Aliluokissa Reset-metodi tulee korvata palauttamaan objekti alkuperäiseen tilaansa
WriteXml	Kirjoittaa tiedot ja tarvittaessa myös skeeman Dataset-objektista
WriteXmlSchema	Kirjoittaa Dataset-objektin XML-muodossa

Taulukko 10, Dataset -luokan julkiset tapahtumat (MSDN 2005 e.)

MergeFailed	Tapahtuu, kun kohteena ja lähteenä olevissa DataRow-objekteissa on sama pääavaimen arvo ja EnforceConstraints-ominaisuuden arvo on tosi
-------------	---

Taulukko 11, DataTable -luokan julkiset ominaisuudet (MSDN 2005 f.)

CaseSensitive	Määrittää, erotteleeko merkkijonovertailu isot ja pienet kirjaimet toisistaan DataTable-objektin sisällä
ChildRelations	Palauttaa DataTable-objektin lapsitaulu-yhteydet kokoelmana
Columns	Palauttaa DataTable-objektiin kuuluvat sarakkeet kokoelmana
Constraints	Palauttaa DataTable-objektin ylläpitämät rajoitukset kokoelmana
Dataset	Palauttaa Dataset-objektin, johon DataTable-objekti kuuluu
DefaultView	Palauttaa taulun räätälöidyn näkymän, joka voi sisältää suodatetun näkymän tai cursorin sijainnin
DisplayExpression	Asettaa/palauttaa lausekkeen, joka palauttaa taulua käyttöliittymässä yksilöivän arvon
ExtendedProperties	Palauttaa kokoelman räätälöityjä käyttäjätietoja
HasErrors	Palauttaa DataTable-objekteissa olevien virheiden määrän Dataset-objektissa, johon taulu kuuluu
Locale	Asettaa/palauttaa merkkijonojen vertailussa taulun sisällä vaikuttavan lokaalitiedon
MinimumCapacity	Asettaa/palauttaa alkuperäisen aloituskoon taululle
Namespace	Asettaa/palauttaa nimiavaruuden DataTable-objektin tallennettujen tietojen XML-muodolle
ParentRelations	Palauttaa DataTable-objektin isätaulu-yhteydet kokoelmana
Prefix	Asettaa/palauttaa nimiavaruuden DataTable-objektin tallennettujen tietojen XML-muodolle
PrimaryKey	Asettaa/palauttaa taulun pääavainkenttinä olevat sarakkeet
Rows	Palauttaa tauluun kuuluvat rivit kokoelmana
TableName	Asettaa/palauttaa DataTable-objektin nimen

Taulukko 12, DataTable -luokan julkiset metodit (MSDN 2005 f.)

AcceptChanges	Hyväksyy kaikki tauluun edellisen AcceptChanges-metodin kutsun jälkeen tehdyt muutokset
BeginInit	Aloittaa lomakkeella olevan tai toisen komponentin käyttämän DataTable-objektin alustamisen
BeginLoadData	Asettaa pois päältä ilmoitukset, indeksoinnin ja rajoitteet tietojen ladattaessa
Clear	Tyhjentää DataTable-objektista kaikki tiedot
Clone	Kloonaa DataTable-objektin rakenteen, sisältäen kaikki skeemat ja rajoitteet
Compute	Laskee annetun lausekkeen arvon suodatusehdon täyttävillä riveillä
Copy	Kopioi DataTable-objektin rakenteen ja tiedot
EndInit	Lopettaa lomakkeella olevan tai toisen komponentin käyttämän DataTable-objektin alustamisen
EndLoadData	Asettaa päälle ilmoitukset, indeksoinnin ja rajoitteet tietojen lataamisen jälkeen
GetChanges	Palauttaa kopion DataTable-objektista, sisältäen latauksen tai AcceptChanges-metodin kutsun jälkeen tehdyt muutokset
GetErrors	Palauttaa taulukon virheitä sisältävistä DataRow-objekteista
ImportRow	Kopioi DataRow-objektin tauluun, säilyttäen ominaisuusasetukset sekä alkuperäiset ja nykyiset arvot
LoadDataRow	Etsii ja päivittää määrätyn rivin. Jos riviä ei löydy, luodaan uusi rivi annetuilla arvoilla
NewRow	Luo uuden DataRow-objektin, jonka rakenne on sama kuin taulun
RejectChanges	Peruuttaa kaikki tauluun muodostamisen tai AcceptChanges-metodin kutsun jälkeen tehdyt muutokset
Reset	Palauttaa DataTable-objektin alkuperäiseen tilaansa
Select	Palauttaa taulukon DataRow-objekteja
ToString	Palauttaa TableName- ja DisplayExpression -ominaisuudet, jos ne ovat määriteltä

Taulukko 13, DataTable -luokan julkiset tapahtumat (MSDN 2005 f.)

ColumnChanged	Tapahtuu, kun DataRow-objektin määrätyn DataColumn-objektin arvo on muuttunut
ColumnChanging	Tapahtuu, kun DataRow-objektin määrätyn DataColumn-objektin arvo on muuttumassa
RowChanged	Tapahtuu, kun DataRow-objekti on muutettu onnistuneesti
RowChanging	Tapahtuu, kun DataRow-objekti on muuttumassa
RowDeleted	Tapahtuu, kun taulun rivi on poistettu
RowDeleting	Tapahtuu, kun taulun riviä ollaan poistamassa

Taulukko 14, DataColumn -luokan julkiset ominaisuudet (MSDN 2005 g.)

AllowDBNull	Asettaa/palauttaa arvon, joka määrää voiko sarake sisältää null –arvoja
AutoIncrement	Asettaa/palauttaa arvon, joka määrää lisätäänkö uuden rivin sarakkeeseen automaattisesti arvo
AutoIncrementSeed	Asettaa/palauttaa lähtöarvon sarakkeelle, jos sen AutoIncrement -ominaisuuden arvo on True
AutoIncrementStep	Asettaa/palauttaa sarakkeen arvon kasvun suuruuden, jos sen AutoIncrement -ominaisuuden arvo on True
Caption	Asettaa/palauttaa sarakkeen otsikon
ColumnMapping	Asettaa/palauttaa sarakkeen MappingType -arvon
ColumnName	Asettaa/palauttaa sarakkeen nimen
DataType	Asettaa/palauttaa sarakkeen tietotyyppin
DefaultValue	Asettaa/palauttaa sarakkeen oletusarvon
Expression	Asettaa/palauttaa lausekkeen, jota käytetään suodattamaan rivejä, laskemaan sarakkeen arvo tai muodostamaan yhdistelmäsarake
ExtendedProperties	Palauttaa kokoelman DataColumn -objektiin liitettyjä räätälöityjä tietoja
MaxLength	Asettaa/palauttaa maksimipituuden tekstille sarakkeessa
Namespace	Asettaa/palauttaa DataColumn -objektin nimiavaruuden
Ordinal	Palauttaa sarakkeen sijainnin DataColumnCollection –kokoelmassa
Prefix	Asettaa/palauttaa XML:n etuliitteen, jota käytetään DataTable –objektin nimiavaruuden peitenimenä
ReadOnly	Asettaa/palauttaa arvon, joka määrää voiko sarakkeen arvoja muuttaa tauluun lisäämisen jälkeen
Table	Palauttaa DataTable –objektin, johon sarake kuuluu
Unique	Asettaa/palauttaa arvon, joka määrää pitääkö sarakkeen rivien arvojen olla yksilöllisiä

Taulukko 15, DataRow -luokan julkiset ominaisuudet (MSDN 2005 h.)

HasErrors	Palauttaa arvon, joka kertoo onko rivillä virheitä
ItemArray	Asettaa/palauttaa rivin kaikki arvot merkkijonon kautta
RowError	Asettaa/palauttaa riville räätälöidyn virheilmoituksen
RowState	Palauttaa rivin nykyisen tilan
Table	Palauttaa DataTable-objektin, jonka skeemaa rivi noudattaa

Taulukko 16, DataRow -luokan julkiset metodit (MSDN 2005 h.)

AcceptChanges	Hyväksyy kaikki riviin edellisen AcceptChanges –metodin kutsun jälkeen tehdyt muutokset
BeginEdit	Aloittaa DataRow –objektin muokkauksen
CancelEdit	Keskeyttää nykyisen rivin muokkauksen
ClearErrors	Tyhjentää rivin virheet, mukaan lukien RowError – ominaisuuden ja SetColumnError –metodilla asetetut virheet
Delete	Poistaa rivin
EndEdit	Lopettaa rivin muokkauksen
GetChildRows	Palauttaa rivin lapsirivit
GetColumnError	Palauttaa sarakkeen virheilmoituksen
GetColumnsInError	Palauttaa jonon sarakkeita, joissa on virheitä
GetParentRow	Palauttaa rivin isärivin
GetParentRows	Palauttaa rivin isärivit
HasVersion	Palauttaa arvon, joka kertoo onko olemassa määriteltyä versiota
IsNull	Palauttaa arvon, joka kertoo sisältääkö määrätty sarake nolla-arvoja
RejectChanges	Peruuttaa kaikki riviin edellisen AcceptChanges –metodin kutsun jälkeen tehdyt muutokset
SetColumnError	Asettaa sarakkeen virheilmoituksen
SetParentRow	Asettaa rivin isärivin

Taulukko 17, DataRelation -luokan julkiset ominaisuudet (MSDN 2005 i.)

ChildColumns	Palauttaa yhteyden lapsisarakkeet
ChildKeyConstraint	Palauttaa yhteyden viite-eheys rajoituksen
ChildTable	Palauttaa yhteyden lapsitaulun
DataSet	Palauttaa DataSet -objektin, johon DataRelation -objekti kuuluu
ExtendedProperties	Palauttaa kokoelman, joka tallettaa räätälöityjä ominaisuuksia
Nested	Asettaa/palauttaa arvon, joka kertoo onko sisäkkäisiä yhteyksiä
ParentColumns	Palauttaa DataRelation -objektin isäsarakeet
ParentKeyConstraint	Palauttaa UniqueConstraint -objektin, joka varmistaa että isäsarakeen arvot ovat yksilöllisiä
ParentTable	Palauttaa DataRelation -objektin isätaulun
RelationName	Asettaa/palauttaa nimen, jolla DataRelation -objekti voidaan löytää DataRelationCollection -kokoelmasta

Taulukko 18, DataAdapter -luokan julkiset ominaisuudet (MSDN 2005 j.)

AcceptChangesDuringFill	Asettaa/palauttaa arvon, joka määrää kutsutaanko AcceptChanges –metodia kun rivi lisätään DataTable -objektiin
ContinueUpdateOnError	Asettaa/palauttaa arvon, joka määrää generoidaanko poikkeus jos rivin päivityksessä tapahtuu virhe
MissingMappingAction	Määrittelee suoritettavan toimenpiteen, jos syötteenä tulevalle datalle ei ole sopivaa taulua tai saraketta
MissingSchemaAction	Määrittelee suoritettavan toimenpiteen, jos DataSet -objektin skeema ei sovi syötteenä tulevalle datalle
TableMappings	Palauttaa kokoelman, joka huolehtii lähdetaulun ja DataTable –objektin välisestä kartoituksesta

Taulukko 19, DataAdapter -luokan julkiset metodit (MSDN 2005 j.)

Fill	Lisää tai päivittää DataSet-objektin rivit vastaamaan tietolähteen rivejä ja luo DataTable-objektin nimeltä ”Table”
FillSchema	Lisää määrättyyn DataSet-objektiin ”Table” –nimisen DataTable-objektin ja muotoilee skeeman vastaamaan tietolähdettä
GetFillParameters	Palauttaa käyttäjän asettamat parametrit SQL Select –lausetta varten
Update	Kutsuu asianomaisia INSERT, UPDATE tai DELETE –lauseita jokaiselle lisätylle, muutetulle tai poistetulle riville määrättyssä DataSet -objektissa

Taulukko 20, DataReader -luokan julkiset ominaisuudet (MSDN 2005 k.)

Depth	Palauttaa nykyisen rivin sisäkkäisyyden syvyyden kertovan arvon
FieldCount	Palauttaa sarakkeiden määrän nykyisessä rivissä
HasRows	Palauttaa arvon, joka kertoo SqlDataReader-objektissa olevien rivien määrän
IsClosed	Palauttaa arvon, joka kertoo onko tiedon lukija suljettu
Item	Palauttaa sarakkeen arvon sen natiivissa muodossa
RecordsAffected	Palauttaa SQL-käskyn suorituksen yhteydessä muutettujen, syötettyjen tai poistettujen rivien määrän

Taulukko 21, DataReader -luokan julkiset metodit (MSDN 2005 k.)

Close	Sulkee SqlDataReader-objektin
GetBoolean	Palauttaa määrätyn sarakkeen arvon boolean-muodossa
GetByte	Palauttaa määrätyn sarakkeen arvon byte-muodossa
GetChar	Palauttaa määrätyn sarakkeen arvon char-muodossa
GetDataTypeName	Palauttaa lähteen datatyyppin nimen
GetDateTime	Palauttaa määrätyn sarakkeen arvon DateTime-objektina
GetDecimal	Palauttaa määrätyn sarakkeen arvon Decimal-objektina
GetDouble	Palauttaa määrätyn sarakkeen arvon double-tarkkuuksisena liukulukuna
GetFieldType	Palauttaa objektin tietotyyppin Type-objektissa
GetFloat	Palauttaa määrätyn sarakkeen arvon single-tarkkuuksisena liukulukuna
GetGuid	Palauttaa määrätyn sarakkeen arvon GUID -muodossa
GetInt16	Palauttaa määrätyn sarakkeen arvon 16 -bittisenä kokonaislukuna
GetInt32	Palauttaa määrätyn sarakkeen arvon 32 -bittisenä kokonaislukuna
GetInt64	Palauttaa määrätyn sarakkeen arvon 64 -bittisenä kokonaislukuna
GetName	Palauttaa määrätyn sarakkeen nimen
GetOrdinal	Palauttaa sarakkeen nimeä vastaavan järjestysluvun

GetSchemaTable	Palauttaa DataTable-objektin, joka sisältää SqlDataReader-objektin sarakkeiden metatiedot
GetSqlBinary	Palauttaa määrätyn sarakkeen arvon SqlBinary -muodossa
GetSqlBoolean	Palauttaa määrätyn sarakkeen arvon SqlBoolean -muodossa
GetSqlByte	Palauttaa määrätyn sarakkeen arvon SqlByte -muodossa
GetSqlDateTime	Palauttaa määrätyn sarakkeen arvon SqlDateTime -muodossa
GetSqlDecimal	Palauttaa määrätyn sarakkeen arvon SqlDecimal -muodossa
GetSqlDouble	Palauttaa määrätyn sarakkeen arvon SqlDouble -muodossa
GetSqlGuid	Palauttaa määrätyn sarakkeen arvon SqlGuid -muodossa
GetSqlInt16	Palauttaa määrätyn sarakkeen arvon SqlInt16 -muodossa
GetSqlInt32	Palauttaa määrätyn sarakkeen arvon SqlInt32 -muodossa
GetSqlInt64	Palauttaa määrätyn sarakkeen arvon SqlInt64 -muodossa
GetSqlMoney	Palauttaa määrätyn sarakkeen arvon SqlMoney -muodossa
GetSqlSingle	Palauttaa määrätyn sarakkeen arvon SqlSingle -muodossa
GetSqlString	Palauttaa määrätyn sarakkeen arvon SqlString -muodossa
GetSqlValue	Palauttaa objektin, joka kuvaa SqlDbType:n rinnakkaismuotoa
GetSqlValues	Palauttaa kaikki arvolliset sarakkeet nykyisestä rivistä
GetString	Palauttaa määrätyn sarakkeen arvon merkkijonona
GetValue	Palauttaa määrätyn sarakkeen arvon sen omassa muodossa
GetValues	Palauttaa nykyisen rivin arvolliset sarakkeet kokoelmana
IsDBNull	Palauttaa arvon, joka kertoo sisältääkö sarake olemattomia tai puuttuvia arvoja
NextResult	Ohjaa readerin seuraavaan tulokseen, kun luetaan tuloksia eräajona suoritettavista SQL-käskyistä
Read	Ohjaa SqlDataReader-objektin seuraavaan tietueeseen

Taulukko 22, SqlParameter -luokan julkiset ominaisuudet (MSDN 2005 l.)

DbType	Asettaa/palauttaa parametrin DbType –ominaisuuden
Direction	Asettaa/palauttaa arvon, joka määrää onko parametri vain asetettava, vain luettava, kaksisuuntainen tai talletetun proseduurin paluuarvoparametri
IsNullable	Asettaa/palauttaa arvon, joka määrää hyväksyykö parametri nolla-arvoja
Offset	Asettaa/palauttaa tasauksen Value –ominaisuudelle
ParameterName	Asettaa/palauttaa SqlParameter –objektin nimen
Precision	Asettaa/palauttaa desimaalien maksimimäärän Value – ominaisuuden esittämisessä
Scale	Asettaa/palauttaa desimaalien määrän, johon Value – ominaisuus pyöristetään
Size	Asettaa/palauttaa kentässä olevan tiedon maksimimäärän tavuina
SourceColumn	Asettaa/palauttaa sen lähdesarakkeen nimen, joka on liitetty DataSet –objektiin ja jota käytetään Value –ominaisuuden latauksessa tai palautuksessa
SourceVersion	Asettaa/palauttaa käytettävän DataRow –version, kun Value – ominaisuus ladataan
SqlDbType	Asettaa/palauttaa parametrin SqlDbType –ominaisuuden
Value	Asettaa/palauttaa parametrin arvon